DOCUMENT RESUME

ED 401 861 HE 029 714

TITLE	New Technology. New Opportunities for Partnering, CAUSE94. Track V.
INSTITUTION	CAUSE, Boulder, Colo.
PUB DATE	95
NOTE	82p.; In: New Opportunities for Partnering. Proceedings of the 1994 CAUSE Annual Conference (Orlando, Florida, November 29-December 2, 1994); see HE 029 709.
AVAILABLE FROM	CAUSE Information Resources Library, 4840 Pearl East Circle, Suite 302E, Boulder, CO 80303 (Individual papers available to CAUSE members at cost of reproduction).
PUB TYPE	Reports - Descriptive (141) -- Speeches/Conference Papers (150)
EDRS PRICE	MF01/PC04 Plus Postage.
DESCRIPTORS	Colleges; *Computer Networks; Computers; Cooperation; Educational Planning; Higher Education; *Information Management; *Information Networks; Information Systems; *Information Technology; Internet; Models; *Partnerships in Education; Productivity; Technological Advancement; Universities
IDENTIFIERS	*Campus Wide Information Systems; *CAUSE National Conference

ABSTRACT

Eight papers are presented from the 1994 CAUSE conference track on new information technology and its uses in higher education institutions. The papers include: (1) "New Tools for Multimedia Development: ScriptX" (Edwin J. Pinheiro); (2) "Providing a Campus-Wide Software Server, or How to Be All Things to All People!," focusing on developments at Arizona State University (Richard Grover and L. Dean Conrad); (3) "Internet Tools Access Administrative Data," which discusses Internet access to administrative and student records at the University of Delaware (Carl Jacobson); (4) "Moving Toward the Virtual University: A Vision of Technology in Higher Education," which examines developments at California Polytechnic State University (Warren J. Baker and Arthur S. Gloster II); (5) "Productivity Tools: An Executive Insight and Evaluation," which discusses the use of computer aided software engineering (CASE) tools at the University of Florida (John E. Poppell); (6) "Designing and Implementing the CUNY Open System Center," which describes the City University of New York's system (Richard F. Rothbard and others); (7) "Taking the Mystery Out of Document Imaging," (E. Leon Daniel and Steve Franzkowiak); and (8) "SIS 2000--The University of Arizona's Team Approach to Client/Server Migration: Strengthening Old Partnerships and Forging New Ones" (John Detloff and others). (Some papers contain references.) (MDM)

****
* Reproductions supplied by EDRS are the best that can be made *
* from the original document. *
****

ED 401 861

New Opportunities for Partnering

CAUSE 94

TRACK V
NEW TECHNOLOGY

Coordinator: Daniel J. Oberst

BEST COPY AVAILABLE

PERMISSION TO REPRODUCE THIS MATERIAL HAS BEEN GRANTED BY

CAUSE

TO THE EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC).

U.S. DEPARTMENT OF EDUCATION
Office of Educational Research and Improvement
EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)
☐ This document has been reproduced as received from the person or organization originating it.
☑ Minor changes have been made to improve reproduction quality.

• Points of view or opinions stated in this document do not necessarily represent official OERI position or policy.

NE 029 714

Proceedings of the 1994 CAUSE Annual Conference

November 29–December 2, 1994
Walt Disney World Dolphin
Orlando, Florida

New Tools for Multimedia Development: ScriptX

Edwin J. Pinheiro
Academic Consultant
IBM Academic Information Systems

Abstract

There are two trends in desktop computing that have become popular: Multimedia and Objects. There has also been one problem that has bedeviled higher education institutions as they create and implement software on desktop computers - the incompatibility between the two dominant standards, Windows and Macintosh. What I propose to present here today is a tool designed to facilitate the implementation of multimedia programs, taking full advantage of object oriented technology, and designed to run on most popular desktop platforms of today, not only Windows and Macintosh, but also OS/2 and UNIX.

ScriptX is a product of Kaleida Labs, a company formed and jointly owned by IBM and Apple Computer. Before delving into ScriptX, let me take a few minutes to justify the use of multimedia in higher education. If multimedia is just a fad without real educational benefit, we are wasting our time and our investment.

New Tools for Multimedia Development: ScriptX

Introduction

There are two trends in desktop computing that have become popular: Multimedia and Objects. There has also been one problem that has bedeviled higher education institutions as they create and implement software on desktop computers - the incompatibility between the two dominant standards, Windows and Macintosh. What I propose to present here today is a tool designed to facilitate the implementation of multimedia programs, taking full advantage of object oriented technology, and designed to run on most popular desktop platforms of today, not only Windows and Macintosh, but also OS/2 and UNIX.

ScriptX is a product of Kaleida Labs, a company formed and jointly owned by IBM and Apple Computer. Before delving into ScriptX, let me take a few minutes to justify the use of multimedia in higher education. If multimedia is just a fad without real educational benefit, we are wasting our time and our investment.

Multimedia Effectiveness

We have all heard clichés such as "A picture is worth a thousand words." Intuitively we agree with such clichés. We know that no matter how well someone describes a face to us, we rarely, if ever, can "picture" the face correctly until we see the person, or a photograph or drawing of the person. We assume that multimedia can somehow enhance the teaching and learning process, but are not quite sure why.

The answer lies, as you would imagine, in the way our brains function. Research done at Woods Hole Oceanographic Institute and other centers of learning has shown that learning is enhanced not only by repetition, but also when multiple senses are involved in the process in a short span of time. Thus if we see and hear something we are more apt to remember it than if we only saw it or heard it. If this something is repeated enough times, physical and chemical changes take place at the synapses in the brain, and the memory becomes permanent.

It is also true that memory is diffused. A memory is not stored in a single synapse, or within a single nerve cell, but over large areas of the brain. In retrieving a memory the brain performs a pattern matching function, trying to match the current stimulus to the memories stored in the brain. The match does not have to be complete. The brain retrieves memories based on partial information. This is why we sometimes confuse a stranger with someone we know. A glimpse of the stranger matches enough of the pattern that the brain retrieves the memory. On closer inspection we find out that the patterns are not a complete match, and we apologize to the person we have just mistakenly recognized. This partial pattern matching is an essential feature in our lives. It allows us, for example, to instantly recognize a dog, although dogs come in a wide variety of breeds, sizes and colors. The richer the pattern of the memory, the easier it is to match. This is why we often form memories by association. A certain smell will invoke, perhaps, memories of childhood or of a loved one. A certain song might have the same effect.

How is this related to multimedia, and specifically to multimedia programs implemented on a personal computer? In several ways:

- Multimedia engages more of the senses. A picture provides a richer pattern than words. Pictures plus sound, moving pictures or animation do the same. Multimedia on the personal computer can be interactive, which engages the higher order thinking processes of the brain, thus facilitating learning.

- Multimedia on a personal computer lets the learner learn at his or her own pace.

- Multimedia on a personal computer lets the learner go over the information multiple times, which also enhances learning.

Although the last two points are not restricted to multimedia programs alone, they are an important factor in successful multimedia courseware. Evidence continues to accumulate on the effectiveness of computers in instruction. Dr. G. Philip Cartwright writes in the 1993 EDUTECH Report: "Do Computers Help Students Learn?"

> "There have been literally hundreds of research studies carried out in which computer based instruction was compared with conventional instruction in a controlled research environment. These summaries reveal that CAI is equal or superior to conventional instruction on the following variables: student achievement, covering both immediate and long term retention; attitude toward both the subject matter and the instructor; and time to complete the task. The generalization is that CAI students realize higher achievement in significantly less time than the conventionally instructed students."

Recently EDUCOM published a document titled "101 Success Stories" which documents many cases in which computers and multimedia have helped the instructional process.

Programs such as Perseus and the Great American History Machine use multimedia not to entertain or to allow mindless pursuit of facts, but to elucidate, explain, and help students develop theories and test them.

Object Oriented Technology

Lets leave the discussion of multimedia for now and look at another promising technology, the technology of objects. While the technology is not new, its benefits had not touched us until programs, such as HyperCard, used some object oriented technology to make developing software easier. Today there are many more recognizable examples of object technology at work: Authoring programs such as Authorware and ToolBook, drawing programs that treat elements of the drawing as objects such as Corel Draw, and even operating systems incorporating object technology such as NextStep and OS/2 Presentation Manager.

Behind object oriented concepts lie many technical terms such as encapsulation, inheritance, polymorphism and instantiation. However it is not necessary to understand the technology to appreciate its benefits. As we explore ScriptX, we will highlight some of the advantages that object technology provides.

Introducing ScriptX

ScriptX is an expression based, interpreted language, which means that every statement you type returns a result. Thus if I type 3 + 4, ScriptX returns the result of the expression, 7. One of the benefits of ScriptX is its interactive nature. I do not have to write a lot of code before I can test it. I can actually run each statement as I type it. Everything in ScriptX is an object, including the numbers I just typed in, and the returned result. ScriptX comes with a wealth of object classes, so I can create a new window as easily as I added two integers. A window is just another kind of object.

Clocks

A unique feature of ScriptX is that one of the object types (classes) that are implemented is clock. Thus you can create clock objects in your program, and assure that events in your program occur at the times you specify, independent of the speed of the machine running the program. You can also synchronize activities in your program and maintain the synchronization, even on a slower machine. As a matter of fact, all of the multimedia objects in ScriptX are derived from the clock class. This makes it easy to start, stop, synchronize, and control the speed of multimedia.

The Foundation Authoring Model

Another interesting architectural feature of ScriptX is the division of function among what are called models, controllers, and presenters. This is termed the Foundation Authoring Model in ScriptX. A model is an object, or set of objects, that we are interested in. A presenter is an object that provides a view of our object on the screen. A controller is an object which controls the interaction between objects in spaces, as well as the interaction between objects themselves.

For example, consider a Text object. It contains a string of text. To show the text on the screen one needs a TextPresenter object. The TextPresenter object handles such issues such as justification, left to right or right to left text display, etc. Consider also a 3D object (these are not currently implemented in ScriptX, but the foundation is there for them). If you wish to show it in two dimensions (for example, the view of a ball would be a circle) you can use a 2D presenter. If you wish to show it in three dimensions, with shading, hidden line removal, etc., you would use a different presenter. The object itself does not change and does not have to be concerned about how it looks. It is the presenter's job to do that. As a final example consider an object that has a temperature property. One presenter can be written to display the temperature as a number, another to represent the temperature as a thermometer-like display, while a third might represent the temperature as a color, so that red is displayed if the temperature is high, and green if it is low, and

shades between the two colors can represent the intermediary temperatures. This division between an object and its properties, and the presenter (how the object is displayed), is very powerful. The objects I construct can be simpler. The presenters can be general purpose. Once I construct a presenter for temperature, I can use that same presenter to display the temperature (or any other linear property) of any object.

The third element of the Foundation Authoring Model is the controller. A controller is an object that controls some aspect of another object's behavior. For example, the Projectile controller gives an object velocity and elasticity properties. The Movement, Bounce and Gravity controllers are able to control one or more projectile objects inside a space. We can, of course, create other controllers as well.

Re-Use of Objects

One of the benefits of object oriented systems is that you can easily re-use objects. Consider the case where there is a portion of a program that performs a function you would like to implement, but is perhaps missing a feature or two you would like, sort of like finding a car on the dealer's lot that is the right make and model, but is missing a desired option or two, or is the wrong color. In traditional programming you muck around with it the code to understand how it works and to incorporate the changes. This has the unpleasant side effect of often introducing bugs in the program, making the code more complicated, and perhaps changing the interface to the code, so that other code that depends on it might have to change as well. You can get rid of some of the problems by copying the code and creating a new piece of code by modifying the copy. Of course, the problem then becomes that your program gets too large, as there are several pieces of near identical code performing separate functions.

On the other hand, in an object oriented program if an object exists that is close in function or appearance to what you desire to implement, you can create a new object that is derived from the original object. You only write code for the new behavior you wish to implement. All of the rest of the object's function is inherited from the original object. There is no danger of adding bugs to the original, working object, there is no duplication of code, and you don't even have to understand how the original object works.

A ScriptX Example

Lets use an example to put it all together. Say we have a number of bitmaps representing the movements of a fish. In order to animate the movement of the fish we need to display the bitmaps in succession at a constant rate. To show a bitmap on the screen I need a presenter, in this case the TwoDShape presenter. To change the bitmaps I will need a clock. Although ScriptX has many object types to support animation, we'll do it the hard way by defining an Animation class (Figure 1). This class will accept a collection of objects, and display them one at a time using the rate of a clock I define. Notice that, in true object oriented fashion, I can pass an Animate object a collection of anything that has a visual representation, not just bitmaps.

I can now run a program and see the fish moving because the animate object is playing the bitmaps in sequence. However, the fish does not move through the water. By creating a new object type, which incorporates the characteristics of the Animation object and the Projectile object (which incorporates the concept of velocity) we can get the fish to move across the window. While I am at it, I would also like to be able to have the fish reappear on one side of the window after it exits the other side, or the animation is going to be short lived. I can create a controller object that will do this. Again, once this object is created, it will take *any* object that falls off the screen on one side and place it on the other side, so this same controller object, which I will call Wrapper, can be used in many other programs.

I would also like to be able to drag the fish in the window and move it to a new position. ScriptX provides a DragController object which works on Dragger type objects. So, somehow, my fish will have to have the characteristics of the Dragger object.

To create my example, then, I define a new class of object called Fish, which inherits the characteristics of:

- Animation (to play each cell in succession).
- Projectile (so it can move in the window), and
- Dragger (so that I can drag it).

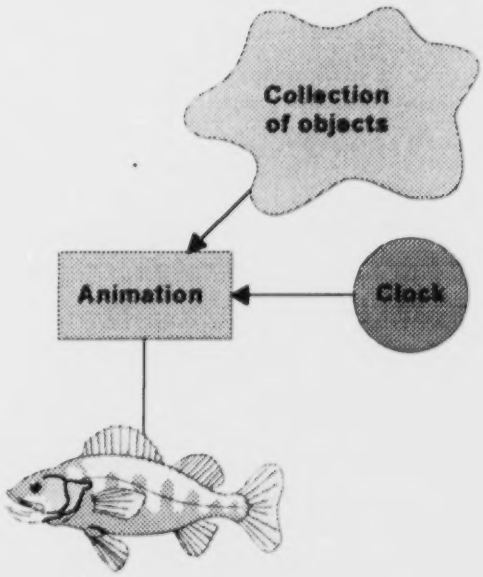


Figure 1: The Animation Object

I will place this object in the window and activate three controller objects to work on the window:

- Movement (to move the fish across the window).
- DragController (to cause the fish to follow the mouse cursor as a result of a drag operation), and
- Wrapper (to move the fish to the edge of the window when it falls off the other edge).

Figure 2 illustrates the program. A new object type (class), called Fish, inherits from the Animation, Projectile and Drag classes. I gave the Fish object the behavior I desire not by writing more code, but by having it inherit the behavior from the other classes.

I also apply the Movement, DragController and Wrapper controllers to the window, so that these controllers will interact with the Fish object to produce the desired behavior. In other words, the Fish object has characteristics that allow it to be moved, dragged, etc. The controllers utilize these characteristics.

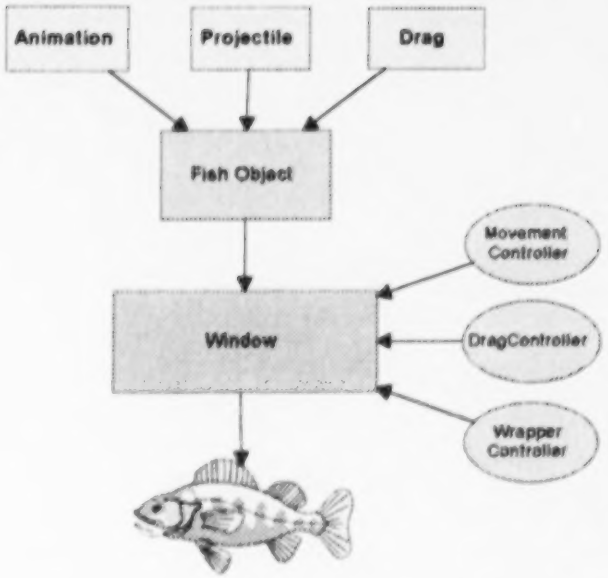


Figure 2: The Complete Program

For example, the Fish object inherits the velocity property from the Projectile class. The Movement controller looks at this property and moves the Fish object by the proper amount at each time interval.

Multimedia

Multimedia is well supported in ScriptX. There is built in support for animation, digital audio, digital video, and MIDI. As previously mentioned, all these object classes are derived from the Clock object, so it is easy to set up and maintain multimedia synchronization. However, multimedia is often file based, and incompatible file formats are a problem for a cross platform product. A PICT file is not the same as a BMP file, a AIFF file is not the same as a WAV file, and a QuickTime movie file is different from an AVI file. ScriptX handles this problem by defining its own internal file formats, and providing importers that will convert the various file types into the internal ScriptX formats. A file may be imported "on the fly," or may be imported ahead of time and saved as an object. Importing a file on the fly can be a time consuming process, so the author of a ScriptX title is likely to do it ahead of time. To handle the situation where a file is not available until run time, such as is the case with a generic media player, ScriptX allows one to invoke an external multimedia player. ScriptX also allows one to manually control multimedia devices via an MCI command interface, and through a similar mechanism on the Macintosh.

Other Features

A vexing problem in object oriented systems is that they normally cannot permanently store the objects they have created. Storing an object is not as simple as writing the object

to a file. The object might reference (point to) many other objects. These references, which are usually implemented as internal pointers, are a problem because when the object is read back at another time the objects might not be in the same place anymore. In C++ all of the objects are built when the program starts, from the code in the program. In other words, a C++ program starts with cookie cutters (class definitions) from which the program creates the necessary cookies (objects). However, objects created by the program as it runs cannot be stored. To solve this situation third party object stores are available. ScriptX, on the other hand, implements its own object store. What this means is that a ScriptX program can start out with the cookies - it does not have to make them. It can also store new kinds of cookies it makes while it runs, so they can be used by other programs. ScriptX uses the Bento architecture (also to be used in OpenDoc) as the mechanism for implementing its object store.

One of the facts of life of an object oriented program is that when it runs, objects are created and others, no longer needed, are destroyed. The programmer does not manage memory as is done in C, where to create, say, a new array, the programmer must invoke the Malloc function to grab some memory, and remember to free it later. The cost of running an object oriented program is that it must include what is called a garbage collector, a part of the program that scans memory and finds objects that are no longer in use and removes them from memory. This is not a simple process, as in object oriented programs objects often reference other objects, so it takes some doing to determine whether an object is to be discarded or not. The problem is that when the garbage collector kicks in, the program ceases to run for a few seconds while the garbage collector does its job. This is of necessity, since the state of objects should not change when the garbage collector is examining them. In most programs this is at most an annoyance. In a multimedia program it is not acceptable. For this reason, the garbage collector in ScriptX (patent applied for) runs on a separate thread in the background and does not cause the program to hiccup. Although it has not been mentioned before, ScriptX is a multi-threaded program, which implements its own thread system so that it can run in single thread environments such as Windows 3.1 and Macintosh System 7.5.

ScriptX supports the Unicode standard. In addition it supports the ISO 10646 standard for representing characters as 4 byte quantities. The storage requirements are minimized, as ScriptX automatically recognizes ASCII characters and stores them as one byte values. ScriptX is also architected to represent text from right to left, and from bottom to top, to be able to render text in several foreign languages and scripts.

The Dynamic Nature of ScriptX

Perhaps one of the most interesting features of ScriptX is its dynamic nature. That means that new objects, say, from an object store, can be added to the system while it is running and these objects will be incorporated into the system dynamically. It is not necessary to recompile the program. Of greater importance is the meaning that the author does not have to plan all of the interactions between objects in the program in advance. New objects can be added later, and they will adapt to the program in much the same way that a child adapts to a new neighborhood.

The potential of this feature is that independent parties can create libraries of objects for ScriptX, in the same way that vendors now create and sell libraries of clip art. One can envision Chemistry objects, Biology objects, Psychology objects, and many others. Thus an author can build his or her own ScriptX program, but populate it with existing objects they did not have to create.

ScriptX Today

Where does ScriptX stand? This fall the ScriptX team at Kaleida has been busy working on footprint and performance issues. Any significant new technology such as ScriptX tends to stress the currently available hardware platforms. This was certainly the case when Windows was first released. Given the rapid advances in platform speeds and capacities, this should not be a problem for long. ScriptX is slated for release on December 16 of this year, initially on both Windows and Macintosh platforms, and soon thereafter on OS/2.

Perhaps a more significant concern for faculty authors is the lack of a visual front end for ScriptX. While ScriptX will ship with tools to translate some aspects of Macromedia Director and Asymetrix ToolBook programs, these are stop gap measures at best. Kaleida hopes that third party developers will develop such tools, in the same way that companies such as Watcom developed Visual REXX for the REXX language. Once such tools are available, ScriptX should provide an exciting, powerful and cross platform enabled multimedia authoring tool. It is also a terrific tool on which to learn the ins and outs of object oriented concepts and programming.

The presentation at CAUSE94 will include demonstrations of live ScriptX code and examples.

Providing A Campus-Wide Software Server
Or How To Be All Things to All People!

CAUSE94 Presentation

Richard Grover
and
L. Dean Conrad

Arizona State University
Tempe, Arizona

Abstract:
Distributed computing systems are a reality, but a wealth of computation hardware is useless without software. Managing and funding the software in a distributed environment is a headache! Faculty need access to the instructional software available in the student computing sites to prepare class assignments. Students need access to the same software to complete their class work. The confinement of software to specific devices or locations contradicts the theme of distributed computing. Everyone needs access to current software to ensure compatibility and to reduce the computing support load. Faculty and staff need access to a diverse set of software, but they do not always have the budget to purchase and maintain this software let alone the time to install it and maintain updates. All this argues for the extension of the student site server concept to a campus-wide software server concept. Without something like this, it will be nearly impossible to adequately manage the burgeoning volumes and diversity of software on our campuses. This paper describes the vision, issues, and estimated costs involved in setting up and running such a service.

Background

As the campus computing environment continues to transform itself from centralized to distributed in nature, one of the key leadership challenges that has emerged is determining how to provide software and support in this environment. Traditionally, student computing sites have provided access to computing hardware and software so that students could complete their course work. To make efficient use of the distributed computing resource available to faculty, they need access, either from their offices or from home, to the same software that is installed in the student computing sites. With personal ownership, or distributed computer facilities, students also need access to this software from anywhere on campus or from home. Many of the clientele of the central Information Technology (IT) department have become sophisticated users of technology. They need access to an increasingly diverse array of current software products that are maintained and compatible with one another and with the campus computing environment.

The costs associated with providing access to adequate software and supporting it, are substantial and--while budgets have improved over the drastic cuts experienced a few years ago--it is unlikely our institution will be able to afford continued proliferation of the present highly distributed and individualized software purchase and distribution schemes. Furthermore, there are significant management headaches involved with supporting a distributed server environment, e.g., backups, restores, off-hours support, network management, and systems administration. All this takes time and resources away from mission-critical departmental activities.

A new kind of centralized management and support scheme is in needed. However, our customers will **not** return to the old days of their forced dependence on the central IT department. Thus a new synergistic balance between independence and centralized economy-of-scale is also needed.

The IT department on our campus already has a significant investment in personal workstation software in the student computing sites and the support infrastructure to install and maintain that software. Like most institutions, we have also deployed a campus-wide network. With the right software management and networking products, this investment can be leveraged to provide a campus-wide software server environment that can make software available to anyone associated with the institution. This technology can also be used to restrict access for a particular set of software to a particular community of interest, such as a single college, department, or class.

Components of the Campus-Wide Software Server

At Arizona State University (ASU), we have put a lot of time and resources into establishing the new distributed computing environment for faculty, staff, and students. We are in a position to capitalize on this investment by leveraging a number of inter-related technology components to establish the campus-wide software server. Not all of these components are entirely in place, but most are at least partially deployed at present. The components we expect to use are identified below.

Distributed computing architecture: ASU has established a distributed computing architecture for the campus called ASURITE ("Developing a Distributed Computing Architecture at Arizona State University", CAUSE/EFFECT, V.17, No. 2, Summer 1994). This architecture specifies a coherent technology environment in which all the components are compatible with one another. This architecture allows us to focus on a restricted set of platforms--i.e., Windows, Mac, and UNIX Motif--in order to make the most of our limited resources.

Adequate personal workstations: Part of the ASURITE architectural definition is the general availability of adequate personal workstations on the desktop. A minimum of a 386 PC or a 68030 Mac is required to run the kind of client software needed. At ASU, the Provost, Dr. Milton Glick, has provided significant workstation "infusion" funding over the past couple years which has essentially brought faculty desktop systems up to par; administrative desktops will be a future priority.

Universal network connectivity: These personal workstations must be connected to the campus network. At ASU, we have deployed a campus-wide Ethernet-TCP/IP network with selected links upgraded to the higher speed FDDI as necessary (and affordable). The same "infusion" funding mentioned above has also allowed a substantial portion of the campus to become networked.

Replace LAN 's With a WAN : In order to minimize the support overhead associated the campus-wide software server, we are planning replacement of our existing Banyan and AppleTalk LAN's in the student computing sites with a WAN. The Andrew File System (AFS) is part of the ASURITE architectural definition. AFS can be used to replace our existing LAN's with a WAN thus unifying our networking support strategy and simplifying training, documentation, and support for the campus-wide software server.

"Dial-up Ethernet": The same functions available on campus must also be available off-campus to foster home access and location independent computing. Clearly, dial-up speeds will restrict some applications with high bandwidth requirements, but most functionality can be provided. We are using the Point-to-Point Protocol (PPP) at ASU to provide this service, although it is restricted to faculty and staff at this time.

Software license management: Products are now available that will allow better management of site licenses in order to minimize the number required for any given package; to avoid a one license per computer approach. These products keep track of, and limit, concurrent usage of a particular software package to the licensed quantities. These management tools can span the campus network to keep track of software use throughout the institution.

Software distribution: The old methods of diskette copying and individual computer installation are time consuming and inefficient. We envision hundreds or even thousands of copies of various software products available via the campus-wide software server. To ensure interoperability among software on the same and separate workstations, it is crucial that there be a way to manage versions of software products. There are two software distribution models: the "push" model and the "pull" model. In the push model, software is distributed from the central location through the network as a universal update. Software is "pushed" out to the customer. In the pull model, software is distributed from the central location when requested by the customer. Software is "pulled" out by the customer. ASU is targeting the pull model for our campus-wide software server.

Campus-Wide Software Server Benefits

We anticipate realizing a number of benefits from establishing the campus-wide software server at ASU including:

- Encouraging student micro purchases. We will never be able to afford to centrally fund all the micros in the sites students will need. The campus-wide software server will encourage student purchase of micros because they will have access to the software they need from home or any other campus location without expensive purchases or having to wait in line for access to a computer in the student sites.

- Faculty access to site software. Faculty will be able to gain access to the same software used by their students. They will be able to develop class demonstrations and assignments from their office or home without having to physically go to a computing site.

- Leveraging of software investments. The investment in software by IT or other entities will be better utilized for a greater return on investment for the institution. When software is not in use on one computer it is available to use elsewhere. This network access can be restricted; for example, software purchased by an academic unit can be restricted to members of that unit.

- A greater diversity of software available to the individual. Products will be available that individual faculty, staff, or students would never be able to afford by themselves.

- Lowered overall software costs for the institution. Fewer copies of software products will be required since everyone will have ready access to a wide diversity of products in the campus-wide software server. Also, more copies of products will be purchased "in bulk" realizing greater volume discounts and a lower cost per copy.

- Lowered overall overhead and support costs for the institution. Much time will be saved by individual faculty, staff, and associated clerical support in navigating the purchasing bureaucracy. Plus all the time required to install and maintain those purchased packages will be saved.

- Location independence. Software will now be available from anywhere rather than being restricted to a particular device, room, or LAN. This reduces unnecessary travel, provides personal convenience, and makes better use of hardware. You don't need to go to the location with the "right" software but can work from any location with adequate computation capabilities and network access.

- Improved ability to support the campus computing architecture. Software product and/or version mismatches are a major headache in maintaining a reliable, responsive campus distributed computing environment. The central IT staff can ensure the products and versions available via the campus-wide software server do indeed work together and are ASURITE compatible.

Strategies for the Campus-Wide Software Server

We have been seeking a unified distribution and license management mechanism. Our investigations indicate that we will likely have to settle for a suite of products, as there does not appear to be a "one-size-fits-all" solution for our three platforms (Windows, Mac, and UNIX Motif). Many of the leading contenders address one aspect, distribution or license management, but not both. Our unified file system, AFS, promises to simplify the distribution issue; however, further development is needed.

While we expect greater efficiency of investment for the university as a whole, a unified licensing effort requires concentration of money into a common fund. We are now managing software across several IT computing sites as a single resource. If we are to transition this IT site resource to a campus resource, we estimate that approximately 1000 additional Windows clients and roughly 500 additional Mac clients will be needed. Extrapolating from our site software licenses, the additional costs could be $250K+ for an initial purchase, with $65K+/year in additional maintenance and upgrade charges. Additional servers will be needed, with an estimated initial cost of $100K and an increase of $10K/year to our maintenance expenses. We

have not yet settled on a funding strategy. Will the additional budget be provided centrally or do we need to sell "subscriptions" to the service in order to fund it?

We are planning a two phase pilot. The first phase is already underway and involves opening the software to a relatively small academic unit, the College of Architecture & Environmental Design (CAED). Macintosh products are available (pull model) to all associated with CAED. This first phase is intended to let us discover technical and administrative issues associated with managing a campus-wide software server.

In the second phase, we intend to broaden the scope to include PC products as well as Mac products and involve a broader base of students and faculty. This phase of the pilot is expected to cost ~$25K and is targeted to begin spring semester '95.

We will need to do a Request For Proposal (RFP) for products that can support the software license management and distribution functions. We have identified the following requirements for the RFP:

- support WINDOWS/Mac/UNIX Motif clients;
- support access that is customer location independent;
- provide usage statistics by application and platform to help track usage and predict the need for additional licenses;
- provide the capability to restrict access to software based on rules (e.g., restrict software usage to certain departments or colleges)
- provide the capability for central and distributed management (license administration);
- support the Kerberos standard for authentication;
- be compatible with AFS;
- support version control for software distribution; and
- provide software distribution functions.

Implementation of a Virtual Software Library at ASU

Earlier LAN-based software management included software distribution and license control functions, but only within the LAN. With the interconnection of several student computing site LANs into the campus network, we began looking for management mechanisms that could span the campus network.

The first package we implemented that could do this was KeyServer, from Sassafras Software, which was implemented to provide the license management along with AppleShare file servers that provided software distribution. A KeyAccess client is installed as an extension to the Macintosh operating system. Software is keyed so that it invokes the KeyAccess client during software launch. The KeyAccess agent contacts the KeyServer to register the launch of the software. As implemented, KeyServer uses AppleTalk protocols for communication between client and server. The license limits are coded in the KeyServer. When all licensed copies are in use, a launch attempt is denied and the person is given the option of waiting in a queue for one of the current users to close the application.

With KeyServer controlling simultaneous launches, it is easy to prove license compliance to software vendors. Keyed software is open to copying from the file server to the local hard disk, but the keyed version of the software cannot launch without KeyServer authorization, so the software is useless off the network. Software control is no longer dependent on restricting distribution, but rather on the active monitoring of simultaneous use.

KeyServer has been running as the software control for the Macintosh systems in the IT sites for over a year now. The distribution of software is still through AppleShare file servers, although we are experimenting with AFS. The software is now managed as a single resource independent of specific locations. A limited license can now be served to any Mac participating in the KeyServer/KeyAccess launch control mechanism. A student can compute at whatever location is convenient, with access to the same suite of software.

KeyServer can also use TCP/IP instead of AppleTalk, which is in keeping with our strategic direction as defined in ASURITE; however, we have not yet shifted to TCP/IP. Sassafras Software has just released a DOS client, thus we are planning to implement KeyServer as the launch control for our PC's as well as our Macs.

The Phase One Pilot

The College of Architecture & Environmental Design has about 1000 students total, which constitutes a little over 2% of the total student body of ASU. CAED has a relatively large installed base of Macintosh computers among students, staff and faculty (for its size). Some key pieces of graphical software are licensed by IT in quantities capable of supporting CAED's needs but were previously only available by traveling to one of the IT sites. With the implementation of KeyServer, CAED purchased a license for the KeyAccess client for its Macs, and is now running software on its Macs from the IT licenses within the same launch control that serves the IT sites.

CAED also has an inventory of limited copies of specialized software of particular interest to the design disciplines. This software is also controlled through the central KeyServer. CAED owned software is no longer available on the limited number of machines that have the few copies of specialized software. Rather, it is accessible to all Macs through the network from a file server. Launch denials are tabulated and used as justification for the expense of limited budget resources on expansion of licenses for those software in highest demand.

The additional usage from CAED has stayed within the existing licenses of IT, demonstrating the ability of the campus-wide software server to make fuller use of software within its license limits. The performance of KeyServer has proven adequate for the larger audience in this initial pilot. We have experienced minor problems with launch approval due to network performance caused by systems other than KeyServer, but the network load of KeyServer activity is negligible.

Challenges

There are still some significant hurdles to overcome in providing a fully configured software server to the ASU community. One is the lack of a PC AFS client which will perform adequately. We deployed AFS as a replacement for our Banyan LAN's in the student computing sites at the beginning of fall semester 1994, but had to go back to Banyan because of sever performance problems with the NFS/AFS translators we were using to bridge the NFS clients on the PC's to AFS. We had similar, problems on the Macintosh computers. We attempted to use an AppleTalk/AFS translator in servers so that the Macintosh clients could use the built-in features of the Mac OS to connect to the AFS servers. Performance limitations have put this strategy on hold. Restricted budget flexibility for this fiscal year may delay the second phase of the pilot.

Conclusion

Distributed computing environments have the potential to create provincialism in the management of software, leading to redundancy of effort, duplication of licenses, limited utility of scare resources, and general inefficiencies in the acquisition and management of software. However, the interconnectivity of the distributed environment coupled with network distribution and control of software promises to overcome the downside of the distributed computing environment for software availability.

We are convinced the campus-wide software server is an idea whose time has come. There simply does not appear to be another viable alternative to manage the burgeoning demand for software on campus. With this concept, software will be recognized as a strategic resource in the distributed computing environment of ASU. In a sense, the distributed computing environment coupled with the campus-wide software server gives units the freedom to do their own thing where appropriate, but to rely on the central organization for those needs held in common with the rest of campus. And allow IT to (finally) be all things to all people!

About the Authors

Mr. Richard Grover is a Support Systems Analyst Principle with Computing & Network Consulting Services at Arizona State University. He has been involved in the support of microcomputer facilities at ASU for 10 years and is presently supporting the College of Architecture & Environmental Design. He had several years of computer experience with geographic information systems and graphics prior to joining the staff at ASU. He has Bachelor of Landscape Architecture from Utah State University and a Master of Landscape Architecture from Harvard Graduate School of Design. He may be reached by phone at 602-965-5654, and by email at Richard.Grover@ASU.EDU.

Mr. Conrad is Director of Computing and Network Consulting Services for Arizona State University with management experience in a variety of technical and applications support roles in both the public and private sectors. He has written and presented on various distributed computing policy, organizational, and technical issues over the past four years. Conrad holds an M.S. degrees in Computer Science from Arizona State University. He may be contacted at (602)965-5620 or via electronic mail at Larry.Conrad@ASU.EDU.

INTERNET TOOLS ACCESS ADMINISTRATIVE DATA

Carl Jacobson
University of Delaware
Newark, Delaware

The University of Delaware provides widespread access to its administrative systems, delivering improved customer service to students, faculty and staff. The Internet's free, public, outreach tools (Gopher and Mosaic) have been merged with the institution's closed, proprietary administrative systems (student records and human resources).

Private, personal information, including student and personnel records, is integrated with the public, general information of the campus-wide information system. Freely distributed clients for DOS, Windows, MAC and Unix workstations allow access to official, production data from both MVS and Unix platforms.

The methods employed to achieve this success are simple, inexpensive and easily adapted.

ADMINISTRATIVE SYSTEMS AND CUSTOMER SERVICE

While the administrative systems of the University can be characterized as closed, proprietary, private, controlled and secure, the student's view of computing is open, pedestrian, public and wide-reaching.

In keeping pace with trends toward increasingly student-centered service, Delaware's administrative systems have been reworked to place an emphasis on self-service. Self-service technologies have been applied to deliver integrated information services directly to the customers in a timely manner. These technologies empower the customer and provide cost-effective, automated services that know no geographic bounds.

THE ROLE OF THE CLIENT-SERVER MODEL

With a healthy portfolio of existing mainframe-based administrative systems, Delaware chose to adapt existing information resources to open network technologies in order to meet the goals of improved customer service.

It is impossible to grant the large, expanding customer base direct access to these mainframe-based information systems. Faculty and research users of "academic" machines have little desire to log on to "administrative" machines and navigate through unfamiliar territory in search of needed information. Nor can 22,000 students be allowed to log on to the mainframe to review grades on the day they are posted.

Closed, proprietary systems must be opened to allow such "pedestrian" use. Administrative information services must be adapted to behave more along the lines of publicly available campus-wide information systems (CWIS).

To meet these goals, Delaware chose to leverage existing resources by merging
-the established, closed, proprietary mainframe-based administrative systems with
-the emerging, open, public, client-server based campus-wide information systems,
in order to
-deliver customer services in the environment of the customer,
-do "administrative things" in "the student way",
-allow the free, public access tools of the Internet to be used to do official university business.

The key to successfully merging these technologies is "compromise". It is necessary to bring the security of the administrative environment to Internet tools, while opening the administrative systems to Internet protocols.

At Delaware, official institutional data is maintained using Software AG's ADABAS database management system and processed by programs written in COBOL and Natural (a programming language), while CWIS information is collected, maintained and delivered using Gopher and World-Wide Web (WWW). Gopher and Mosaic use is widespread among our campus customers, while our Natural/ADABAS systems are robust and useful. These resources have been combined in a unique way to quickly deliver improved information services to student, staff and faculty.

Transforming the "host" of a host-terminal system into the "server" of a client-server system allows the application of technologies geared to improving customer service. The host and its associated applications becomes part of a client-server network enabling outreach and supporting diverse data types.

OPENING CLOSED SYSTEMS

The client-server model of computing makes the opening of such closed systems a simple task. The work required focuses on the need to create a "server" that speaks an open protocol on the user side (Gopher protocol in this case) and understands and interprets the proprietary administrative systems on the database side (ADABAS, Natural, and COBOL). Gopher and Mosaic clients recognize such a server as a Gopher service while the ADABAS DBMS speaks to such a server using existing COBOL or Natural programs.

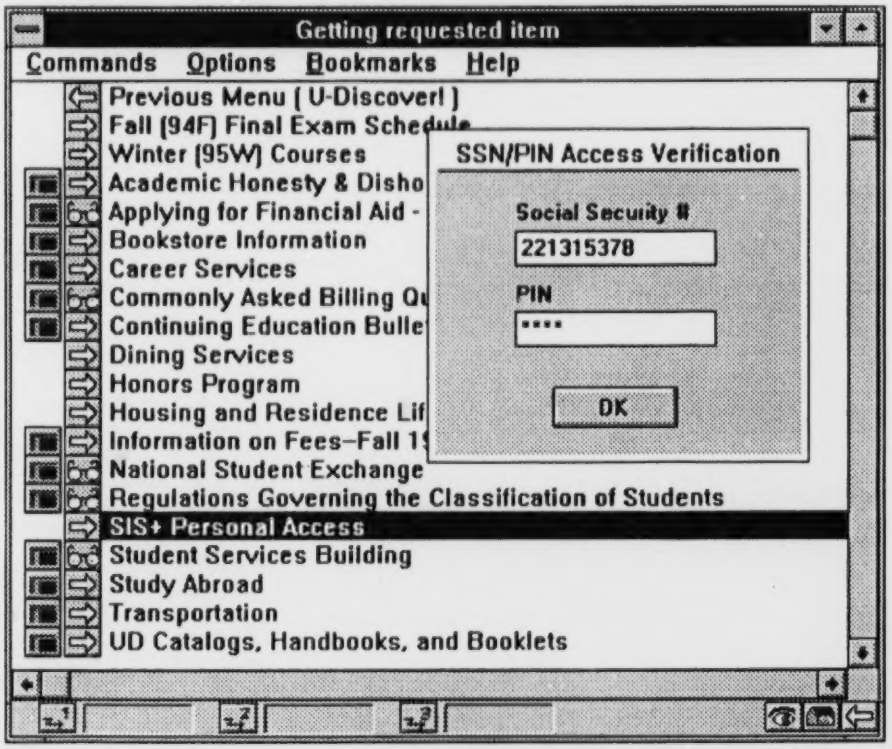
With Gopher and Mosaic clients already in the hands of students, faculty and staff, the issues of training, support and software distribution are minimized. Student grades and transcripts may be accessed in a manner familiar to all existing Gopher and Mosaic users allowing students to use these tools to conduct institutional business as well as to explore academic frontiers.

Since these tools are free and widespread, client-side costs were kept to a minimum. However, in order to provide the levels of security needed in conducting personal business, authentication routines need to be added to these tools. At Delaware, this was first done by adding an encrypted authentication scheme to Gopher clients.

AUTHENTICATION AND AUTHORIZATION

With an overall design goal of "using existing resources whenever possible", SSN/PIN authentication and authorization schemes used for touch-tone registration were enlisted to provide similar security to the Internet clients. PINS (Personal Identification Numbers) were already known and used by students and staff. PIN-based authorization tables were already in place in administrative systems. Therefore, Gopher clients were modified to prompt for SSN and PIN. These values were encrypted and appended to standard Gopher packets to be unpackaged and handled by server-side authorization routines.

Of course, this approach requires that Gopher source code be available. At the time of this phase in development, there were very few Gopher clients, Mosaic was unavailable, and Gopher source code was fairly easy to come by. Since that time, the construction of Internet "browsers" has become a growth industry and there are now many Gopher and Mosaic clients to choose from, and source code has become hard to come by.



SSN/PIN Authentication Added to Gopher Clients

In keeping with the spirit of "doing it the student way", it is important to provide access to users of any Gopher or Mosaic client, on any platform. To meet this goal, a second version of the client was developed with the SSN/PIN authentication and encryption routines "externalized" and packaged as a MIME (Multi-purpose Internet Mail Extensions) viewer. This viewer was written using a cross-platform development tool to allow one piece of source code to be the basis for viewers for DOS, Windows, Mac an Unix users.

In this way, users of any MIME compliant Gopher or Mosaic clients may define the University of Delaware SSN/PIN viewer, called "You-View" to handle any requests requiring University of Delaware authentication. The current authentication scheme uses a single key encryption algorithm and includes the encryption of the Internet station address to guard against the rebroadcast of clandestine packets. With the implementation of a campus-

wide Kerberos authentication service, its expected that Kerberos will eventually replace and improve this current scheme.

While the SSN/PIN "key" is protected using encryption, all text is returned in unencrypted format. As the popularity of "You-View" continues to grow, plans are underway to eventually "garble" this text using DES (Data Encryption Security) encryption to add an additional level of security to the entire process.

The use of Gopher ASK blocks and Mosaic forms to prompt for SSN and PIN was considered during development of "You-View". At this writing the results of either method are transported across the network in unencrypted formats and key encryption is required on our campus. However, with the development of secured Mosaic and Gopher clients for business and commerce comes the possibility of using off-the-shelf Gopher and Mosaic clients to provide secure, encrypted authentication. These developments hold great hope for the future, when Internet tools will be routinely applied to deliver campus business services.

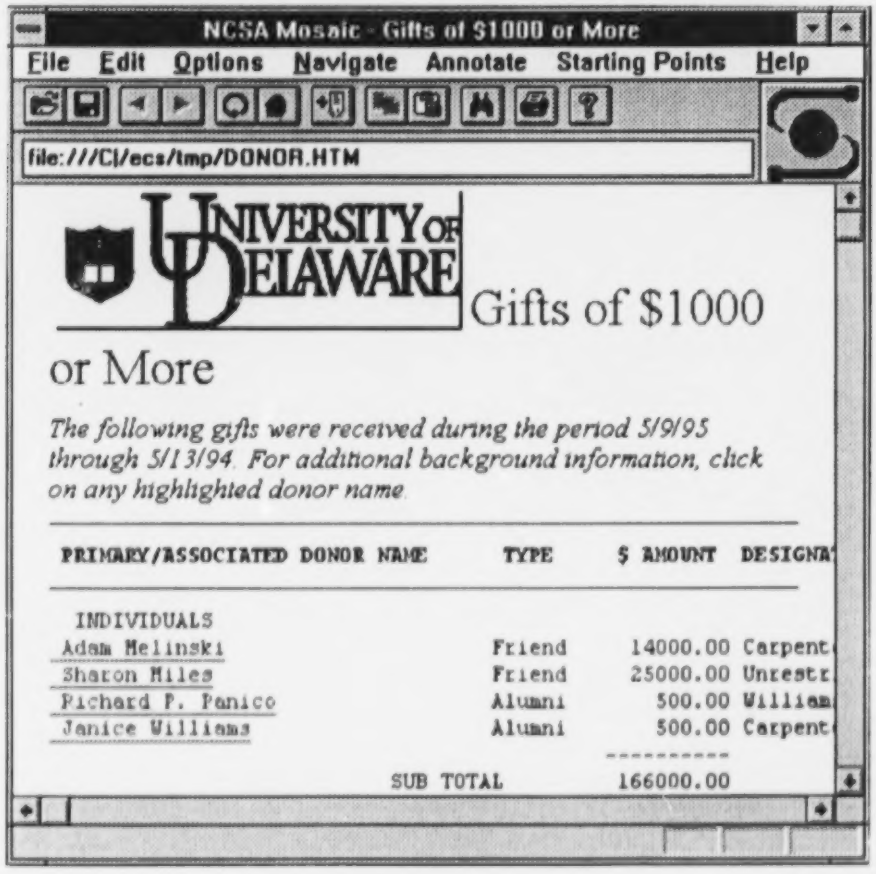
SERVERS PROVIDE SERVICE

On the service side, it is necessary to translate Internet protocols into the languages of our administrative systems and databases... to provide a bridge between the Internet protocols of gopher, WWW and email and the administrative 3GLs, 4GLs and DBMSs.

With administrative systems residing in an MVS mainframe environment, it was necessary to write a "server" to run in this environment, accept IP packets, recognize Gopher protocol, and call administrative application programs based on the content of these Gopher packets. Gopher and Mosaic can routinely display directories and text. Application programs already existed on the mainframe to produce transcripts, grade reports and schedules as text reports. Instead of printing these reports, the server needs only to package them as Gopher replies and send them back out onto the IP network.

Again, this effort was completed before the advent of Mosaic, so that text is packaged in Gopher format. Currently there is no need to hyper-link items in the student or personnel reports, so these documents have not converted to the HTML (HyperText Markup Language) format used by Mosaic. Unlinked documents allow all text to continue to be used by both Gopher and Mosaic clients. However, the generation of HTML documents is appealing and would be useful in developing applications such as Internet-based Executive Information Systems (EIS).

A mainframe-based HTML server has been developed at Delaware as a "proof-of-concept" trial, generating hyperlinked management reports that are delivered via MIME-compliant email. This would allow university management to receive regularly generated summary reports with built-in "drill-down" capabilities and links to official, production data from live administrative databases or links to more diverse data-types such as photographic or document images.

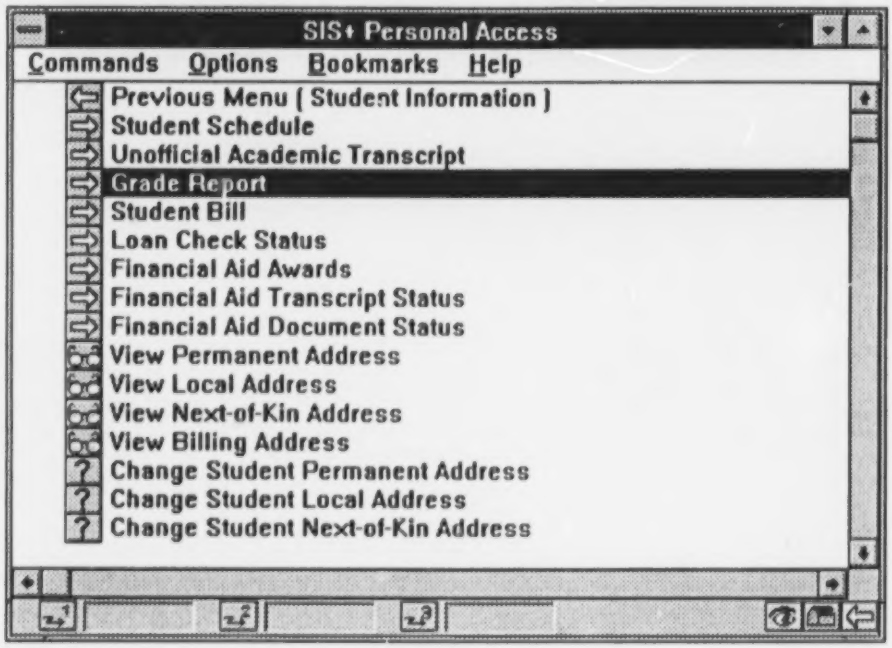

NCSA Mosaic - Gifts of $1000 or More
File Edit Options Navigate Annotate Starting Points Help
file:///C|/ecs/tmp/DONOR.HTM
UNIVERSITY OF DELAWARE
Gifts of $1000 or More
The following gifts were received during the period 5/9/95 through 5/13/94. For additional background information, click on any highlighted donor name.
PRIMARY/ASSOCIATED DONOR NAME TYPE $ AMOUNT DESIGNA
INDIVIDUALS
Adam Melinski Friend 14000.00 Carpent
Sharon Miles Friend 25000.00 Unrestr
Richard P. Panico Alumni 500.00 William
Janice Williams Alumni 500.00 Carpent
SUB TOTAL 166000.00

Hyperlinked Executive Summary Report

STATELESS CLIENT-SERVER RELATIONSHIPS

A significant advantage to adopting a Gopher-like server to provide student services lies in the "statelessness" of Gopher and WWW servers. The transactions may be viewed as "stateless" in that a server has no lasting connection with each requesting client. The server "comes alive" upon receiving a request message across the network, interprets and fulfills the request by passing a message back across the network and returns to a "wait state" until the next user request comes along.

Students do not log on to the administrative system, there is no datacommunications overhead. A single started task monitors an Internet port and responds to customer requests. This "stateless" client-server relationship allows many customers to effectively use administrative resources without becoming members of that environment.

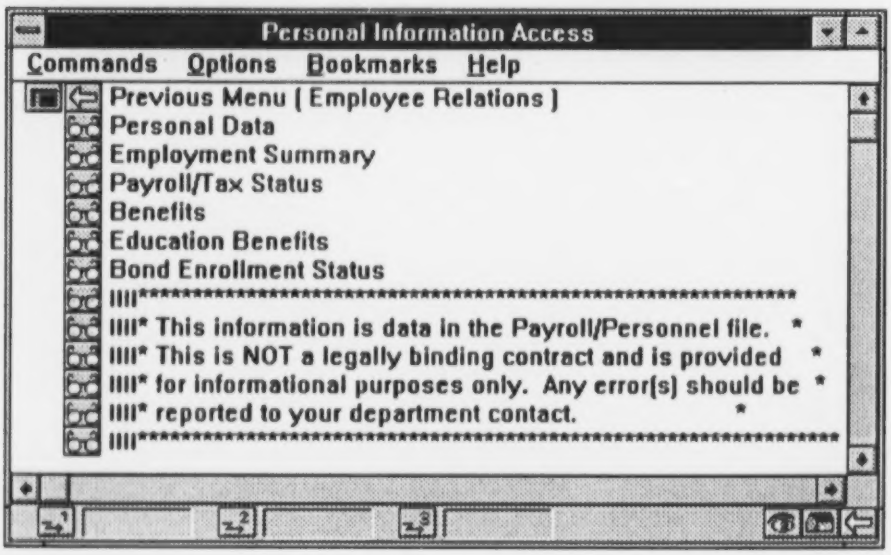


Menu of Student Services

Without the overhead of CICS or TSO sessions the server performs its simple tasks with little impact to the overall system. Response is immediate, even for longer packages, such as transcripts. In addition, due to the nature of the current breed of Internet tools, the response time expectations of Gopher and Mosaic users are lower than those of interactive transaction-based systems, so that if a delay is encountered it is unremarkable.

Besides the great advantage of using existing programs to produce grades and transcripts and schedules, this "interpretive server" has the advantage of accessing production data directly. It does not rely on data extracts but instead returns timely and accurate information from the official, production records of the institution. As students perform touch-tone drop-add, they can immediately confirm schedule changes. As students pay bills, they can quickly print summaries of charges and payments. With many business transactions reaching the database in real time, it has become necessary to report the changes in real time. "Just-in-time" production of course schedules and transcripts calls for this level of timeliness. The stateless, interpretive server allows this to be accomplished easily and inexpensively.

With interpretive servers speaking to administrative programs, existing tasks, such as transcript production, can be reused rather than re-developed. Upon request from a student client, the server simply invokes the existing COBOL transcript program, however, instead of printing or displaying the results, they are packaged in a Gopher packet and sent it out onto the network.


Personal Information Access
Commands Options Bookmarks Help
Previous Menu (Employee Relations)
Personal Data
Employment Summary
Payroll/Tax Status
Benefits
Education Benefits
Bond Enrollment Status
IIII**********
IIII* This information is data in the Payroll/Personnel file. *
IIII* This is NOT a legally binding contract and is provided *
IIII* for informational purposes only. Any error(s) should be *
IIII* reported to your department contact. *
IIII**********

Menu of Personnel Services

Servers have been written to run on both MVS and Unix platforms to allow information to be gleaned from various databases across campus and to take advantage of the relative merits of both operating systems.

THE CUSTOMER IS THE CLIENT

At Delaware the "You-View" client was deployed as a "self-service" technology. It was first made available at character-based, public kiosks, then to public computing sites, labs and libraries. After all residence halls were wired the same client programs used to deliver services to these sites were used to delivery services to individual students in their rooms. For some time only these "stateless", workstation clients could be used for personal access to student and personnel information. The clients running on central time-share systems were not enabled with the SSN/PIN authentication. This was done to allow the stateless clients to

gain a foothold in the network and to avoid the possibility of 22,000 students logging onto a timeshare system the day the grades are posted.

Now that the clients have been established, the Unix-version of "You-View" will be placed on a central cluster of time-share machines heavily used by students. This will allow increased access to all "You-View" student services.

TOUCH-SCREEN, MULTI-MEDIA KIOSKS

Public access workstations at many locations across campus, including all public computing labs, computing classrooms, business service centers, libraries and residence hall lounges, provide access for those who do not have their own workstation or network connection.

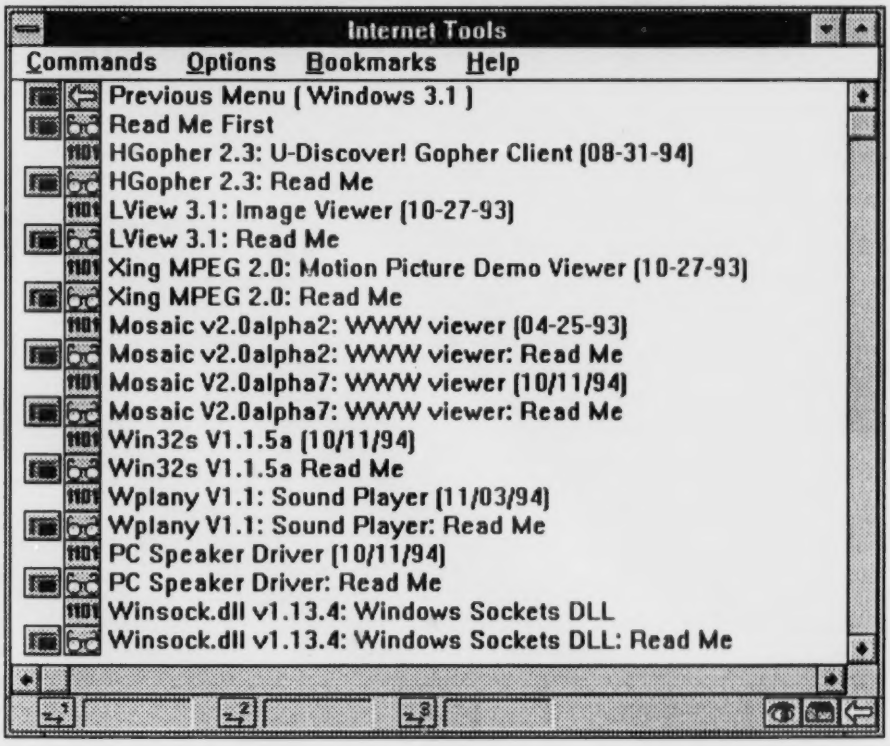
A multi-media authoring tool developed by a University of Delaware professor, has been made "Internet aware" allowing it to speak Gopher protocol. This tool, originally targeted as a classroom technology, is now used to develop compelling, multi-media, touch-screen kiosk applications; merging images, sound and video with administrative information.

This object-oriented toolset allows kiosks to be built quickly and inexpensively using existing Internet resources. Thus, the "dusty old" COBOL transcript program is given new life, delivering up-to-date transcripts to users of DOS, Windows, Mac and Unix workstations, Unix dial-in and network users, as well as to customers of self-service touch-screen kiosks located on campus.

SOFTWARE DISTRIBUTION

One advantage of the client-server model of computing is the increased functionality provided at the desktop. Not only can Internet browsers retrieve grades and course schedules, but they can also retrieve and display images, sounds, and even brief video clips. Any "digital object" of reasonable size can be delivered to any client workstation. This includes the delivery of client software itself.

In keeping with the goal of "self-service", Delaware's Internet client software is stored on a Gopher server and made available to anyone in the campus community across the network from Gopher or Mosaic pages. A simple point-and-click causes the newest version of a program to be loaded, across the network, to the user's hard drive.


Internet Tools
Commands Options Bookmarks Help
Previous Menu (Windows 3.1)
Read Me First
HGopher 2.3: U-Discover! Gopher Client (08-31-94)
HGopher 2.3: Read Me
LView 3.1: Image Viewer (10-27-93)
LView 3.1: Read Me
Xing MPEG 2.0: Motion Picture Demo Viewer (10-27-93)
Xing MPEG 2.0: Read Me
Mosaic v2.0alpha2: WWW viewer (04-25-93)
Mosaic v2.0alpha2: WWW viewer: Read Me
Mosaic V2.0alpha7: WWW viewer (10/11/94)
Mosaic V2.0alpha7: WWW viewer: Read Me
Win32s V1.1.5a (10/11/94)
Win32s V1.1.5a Read Me
Wplany V1.1: Sound Player (11/03/94)
Wplany V1.1: Sound Player: Read Me
PC Speaker Driver (10/11/94)
PC Speaker Driver: Read Me
Winsock.dll v1.13.4: Windows Sockets DLL
Winsock.dll v1.13.4: Windows Sockets DLL: Read Me

Software Distribution Menu

UPDATES

Delaware's EZForms electronic forms system is available as a MIME "viewer" enabling protected-field forms to be delivered to Gopher or Mosaic clients. This allows functions such as "change of address" to be performed under the protection of SSN/PIN authentication.

The EZForms application provides automated control of the document routing and approval process and allows users of any campus mail system, on any operating platform to participate as "submitters" or "approvers".

SUMMARY

The interest of the Clinton/Gore administration in a National Information Infrastructure, coupled with the emergence of compelling Internet applications such as Gopher and Mosaic, has contributed to the recent explosive growth of the Internet. Advances in the tools of the national network will impact the processes of teaching, learning and research on our campuses. Many of these same advances will contribute to the way we conduct business and affect daily campus life for students, employees and visitors.

By combining current tools and technologies, existing resources can be re-used effectively to return immediate benefits against small investments. Early adopters of these technologies and methods will gain valuable experience and insight into the issues of delivering networked services and will establish a foundation for controlled growth and change.

As administrators on the Information Super Highway, we will need to move quickly and carry little baggage. From this point on, there will be few rest stops, and little time for planning long journeys. As explorers on the old frontiers of our country had little idea what was in store for them as they journeyed west, we cannot predict the events on the information frontier well enough to lay complex and concrete plans. We cannot contribute to any plan, until we simply begin the journey.

To achieve business and academic advantage, we cannot delay in identifying and applying the emerging technologies of networked information.

-Technologies that recognize the changing nature of proprietary systems, and acknowledge the role of openness in the future success of our networked campuses.

-Technologies that depend on the client-server model of computing to coordinate the deluge of dispersed information events across these networked campuses.

-Technologies that fundamentally change the way we conduct business, our approach to the educational process, and daily campus life.

-Technologies that allow continued improvement of service; enabling the re-engineering of business processes, facilitating client outreach and self-service, advancing teaching and research, and enriching campus life.

Moving Towards the Virtual University: A Vision of Technology in Higher Education

by Warren J. Baker

and Arthur S. Gloster II

Abstract

California Polytechnic State University, San Luis Obispo, is exploring several cost-effective technology solutions aimed at improving learning productivity, reducing labor intensity, and providing new ways to deliver education and better services to students while enhancing the quality of instruction. Strategic planning and partnerships have been key to their progress to date.

Moving Towards the Virtual University: A Vision of Technology in Higher Education

by Warren J. Baker and Arthur S. Gloster II

California Polytechnic State University, San Luis Obispo, is exploring several cost-effective technology solutions aimed at improving learning productivity, reducing labor intensity, and providing new ways to deliver education and better services to students while enhancing the quality of instruction. Strategic planning and partnerships have been key to their progress to date.

After decades of promises based on overhead projectors, video distribution, and other instructional technologies, the ability to improve instruction using information technology has now become a reality. By incorporating a wide range of digitized media into the myriad of curriculum-related activities fundamental to teaching and learning, the quality of both can rise.

A paradigm shift is taking place in higher education instruction, from a mode of faculty–student interaction occurring in fixed locations at specified times to one in which students can access the same instructional resources in a variety of forms, regardless of location, at their convenience. This is possible because several technologies have matured, supporting major changes in how instruction can be delivered to students on the campus, in their homes, or in their work places.

Escalating costs, declining support, increasing demand, and diverse demographics have placed significant pressures on higher education to become more productive. Careful analysis shows that the productivity improvements required cannot be achieved by increasing the workload of the faculty; in fact, any significant movement in this direction will only decrease the quality of instruction. There is simply no room left in the workday of a faculty member to teach more students. Rather, the focus for productivity improvement must be on learning resources that will improve retention and decrease the time needed to earn a degree.[1]

It is this realization that is leading to the paradigm shift towards an instructional model in which students gain access to information resources, faculty lectures and demonstrations, library and research materials, and conferencing and tutorials over networks from digital information organized in servers by the faculty. Students and faculty can "talk" electronically whenever they like. Assignments can be given and received electronically. Faculty can hold "virtual" office hours, freeing them from rigid schedules, and enabling students to obtain information with little waste of time and without sacrificing the fundamental, close-knit quality of the student-mentor relationship. In this developing model, faculty can become facilitators and guides for individual learners rather than simple conduits for transmitting information.[2]

Productivity gains can occur in greater retention, more efficient use of the student's time, easy access to group study over networks, better feedback to faculty, and organized self-assessment and self-pacing. Faculty and traditional classrooms are not replaced, but another dimension is added that greatly improves the efficiency of learning. Studies have shown that students supported by technology-mediated instruction required about one-third less instructional time than students using traditional lecture/textbook methods. Not only did college students using technology learn faster, six months after completing their studies, they tested better on the subject than their peers who had been taught in traditional settings.[3] Other studies have shown that people reluctant to speak in a group are often less inhibited by electronic communications. By increasing opportunities for interaction and participation, electronic scholarship offers a whole new range of pedagogical techniques with which to reach people who have been left out.[4] As this new process of using technology to improve learning

develops, more students at every level, from elementary student to adult learner, will be able to take advantage of this type of instruction.

Technological advances to deliver entertainment or "video on demand" are progressing rapidly. The opportunity exists today to take that technology and apply it to education to overcome economic, cultural, and physical barriers to learning facing the nation as a whole, including continuous retraining of the workforce. This will require colleges and universities to mirror business and industry by delivering "just-in-time" rather than "just-in-case" education, and to pursue cooperative efforts with the private sector to achieve this vision.

California Polytechnic State University, San Luis Obispo (Cal Poly) is exploring several cost-effective technology solutions aimed at improving learning productivity, reducing labor intensity, and providing new ways to deliver education and better services to students while enhancing the quality of instruction. This article shares Cal Poly's experiences to date in creating a vision and plan to develop the infrastructure needed to transform the way education is delivered, presents steps that have been taken or are about to be taken to implement that vision, and details some of the many partnerships that have contributed to the plan's success thus far.

Strategic plans, goals, and issues

Since the mid-1980s, when the University decided to upgrade its administrative computing systems, Cal Poly has aggressively pursued the use of information technology to transform educational services. By the early 1990s, strategic plans for an integrated, online administrative system (OASIS), voice-response registration, online library services, improved telephone service, a campus-wide fiber optic data network, and instructional access to UNIX had all been realized.[5]

Two years ago, Cal Poly's computing advisory committees embarked on another strategic planning effort to define the future role of technology in support of the University's instructional program. This effort coincided with a campus-wide reassessment of the University mission and academic calendar, adoption of a new strategic plan for the campus, CSU system-wide initiatives for using technology to support instruction (see Project Delta sidebar), and a decision to upgrade the central mainframe.

This planning effort was led by the University's Information Resource Management Policy and Planning Committee (IRMPPC) and the Instructional Advisory Committee on Computing (IACC). The IACC includes one faculty member from each of the University's six academic colleges, and representatives from the library, student association, and academic computing services. The IACC chair acts as liaison to the Academic Senate on instructional computing issues and also serves on the IRMPPC along with several faculty members and vice presidents, the library dean, an academic dean, a student representative, and the chair of the Administrative Advisory Committee on Computing.

After consulting with their respective college computing committees, academic departments, the Senate, and other constituency groups, the IACC produced a strategic plan outlining four major goals for academic computing:

- a networked instructional environment, based on universal electronic mail, shared information resources, and computerized classrooms;
- easy access to workstations and networked information services;
- institutional support for faculty and student development of computer-based communication skills; and
- simplified interfaces, procedures, and documentation for accessing networked information services.

The vision that emerged recognizes that technology can benefit learning when it (1) allows a student to take a more active role, (2) allows a teacher to express the content of a course in more than one format, (3) broadens the array of resources brought to a classroom or the student's workstation, (4) increases the opportunities for interaction between teacher and student and for interaction among students, (5) reduces barriers to University services, and (6) increases the productivity of those who support the learning environment.

As envisioned by the IACC, this "next revolution" will cross all disciplines, especially those which have not traditionally used computing in the past, and will emphasize content development, easy access, and information sharing, rather than focusing on the technology itself. Beyond the obvious need for technology enhancements, the IACC strongly

recommended providing incentives and support to enable the faculty as a whole to develop the necessary skills and methodologies to conduct and publish research, create and deliver lectures, and interact with students in this new environment. Other policy/support issues included:

- considering professional development in the technology area when evaluating faculty for retention, promotion, and tenure purposes;
- supporting faculty with well-defined projects for experimenting with new technologies and innovative ways of employing them in the teaching, learning, and research processes; and
- providing instructional designers and technical support to assist faculty in developing content and integrating technology into the curriculum.

In addition, a number of infrastructure issues were identified:

- adequate network connections to faculty offices and classrooms;
- network ports for students to connect portable computers;
- adequate network access from off-campus sites or residences;
- appropriately configured workstations;
- classrooms equipped with systems for displaying prepared lecture materials and sharing information resources; and
- online search and retrieval tools with graphical user interface.

The IACC plan was generally accepted by the faculty, despite reservations by some as to how it would be achieved technically, and what the impact might be on University resources and faculty workloads.

Implementing the vision: a MegaServer approach

After receiving the plan, the IRM Policy and Planning Committee began an intensive study of how to implement the vision. They spent several months analyzing the capacity of existing resources to support the vision and considering various alternatives before recommending going ahead with a plan to develop a multimedia "MegaServer" as part of the planned mainframe upgrade for the campus.

This MegaServer will provide faculty and students with on- and off-campus access to a full range of information technology resources (voice, data, video) in an integrated, networked educational environment. It will also facilitate local and statewide access to full-text articles and publications, electronic library services, databases, and digitized instructional materials, including slides, graphics, and full-motion video. It will also serve as an important node in a client/server arrangement, supporting campus-wide administrative services and functions.

Cal Poly envisions using this MegaServer approach to support its concept of a "virtual university" (see Figure 1), with many potential applications (see sidebar next page). The benefits for the University include (1) improved access by students enrolled in traditional programs offered by Cal Poly, (2) increased access to academic programs by non-traditional students, (3) better prepared students in K-12 and community college programs, (4) improved effectiveness in uses of limited human, program, and financial resources, (5) new revenue streams to offset infrastructure and operating costs, and (6) incentives for faculty to develop new educational materials.

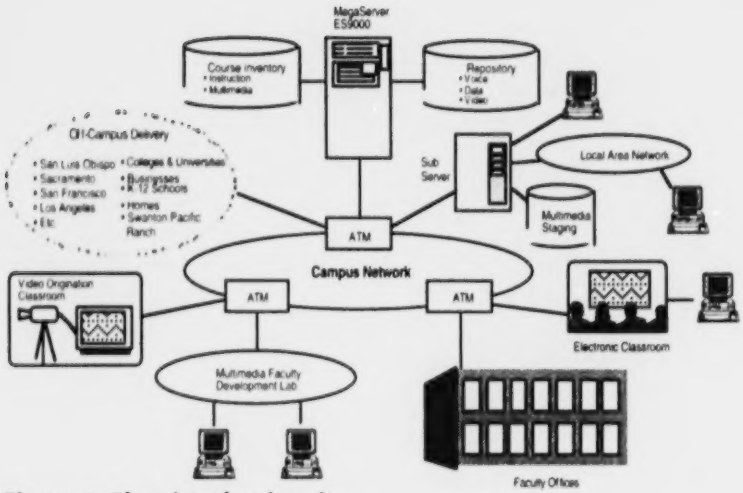


Figure 1: The virtual university

Progress to date

Cal Poly has already taken a number of steps to begin preparing for the virtual university.

In May 1992, the University began using two-way interactive video to deliver courses on campus, between the campus and its satellite agricultural facility 175 miles away, and to the Lucia Mar School District just 20 miles away.

The Faculty Multimedia Development Center (FMDC) was established in March 1993 to provide a variety of hardware, software, and consulting assistance to encourage and support faculty interested in developing and integrating materials into their courses or for delivery over the network. This facility is described in greater detail below in the discussion of support systems.

In September 1993, the University entered into a joint development agreement with IBM to develop and test the MegaServer concept, installing an IBM ES/9000-732 mainframe, LAN File Server/Enterprise System Architecture software, multimedia development workstations, disk storage, and other basic system components. As of spring 1994, the mainframe supported eight concurrent multimedia video streams or sessions to multimedia workstations in the FMDC and a specially equipped classroom. The MegaServer currently supports token ring network access, but most faculty offices and instructional facilities are now or will be equipped with Ethernet connections; extending full-motion video network access to Ethernet connections is a high priority in 1994.

Currently several classrooms are equipped with large-screen video projection systems, Macintosh and IBM-compatible computers or interfaces, and network connections to the mainframe. The University is committed to developing "electronic classrooms" equipped with high-resolution projectors, quality audio systems, and microcomputers with high-speed network access to the MegaServer. With the implementation of network-connected classrooms and the FMDC, faculty can already develop multimedia lectures in the FMDC, store these lectures on the MegaServer, then walk into an electronic classroom, log on to the MegaServer, and retrieve the same lecture for delivery to the students.

Limited resources will make it difficult to equip classrooms quickly enough to meet the anticipated demand for integrating multimedia into the classroom. (Equipping just one such classroom can cost more than $150,000.) To minimize costs and maximize flexibility, the University purchased several laptop computers (at $3,000 –$5,000 each) and portable multimedia-enabled graphics projectors (at $6,000 each) as an interim solution. This equipment can be checked out by faculty to create and deliver multimedia courseware in their office or any classroom. These initial efforts are introducing the campus community to the possibilities of the virtual university by allowing faculty to develop and use multimedia course materials while the MegaServer infrastructure is being more fully developed.

During winter quarter 1994, the University taped its first series of lectures for a course being developed by two faculty members in architecture and construction management.

BEST COPY AVAILABLE

Lectures were taped in the campus video production studio, then rebroadcast over the campus television distribution system during the day and to campus residence halls at night. Students were able to view the lectures at set times or check out tapes of individual lectures to view at home, and to communicate with the instructors during office and lab hours and through electronic mail.

Cal Poly plans to develop the capacity to videotape and "digitize" entire lectures, which can then be edited, indexed, and stored on the MegaServer along with course materials. Both the lectures and materials can be retrieved later to supplement existing classroom instruction, or delivered as "on demand" courses in non-traditional settings, such as a graduate-level degree program for students who work full-time.

To digitize and store lectures on the MegaServer for "on demand" retrieval will require higher bandwidth than is presently available on the campus network. To provide this bandwidth, this summer the University is beginning to beta test an asynchronous transfer mode (ATM) network. IBM is providing optical storage, telecommunications technology, wireless LAN technology, and other support as needed to fully test delivery of full-motion video over the University's fiber optic backbone network. The FDDI hubs will be replaced by ATM hubs capable of using the existing fiber. The FDDI hubs will be recycled and used as routers on the network. In addition, the delivery of interactive video from the MegaServer to four other remote CSU campuses will be tested later this year.

The University is also experimenting with providing on-campus network ports ("docking stations" and "port replicators"). This will allow students to use their own laptop computers to access the network, high-resolution displays, and specialized resources.

Creating a support system

In conjunction with the Cal Poly/IBM MegaServer joint study project, the University established a new management-level position, director of multimedia development, to facilitate the use of the MegaServer and multimedia technologies to deliver education. Since July 1993, the director has concentrated on training faculty, developing instructional content, and coordinating and facilitating efforts by faculty to integrate information technologies into the curriculum. To date, nearly 100 faculty members have completed training or sought individual consultation, while another twenty have been helped with specific multimedia projects. IBM is also providing support to help faculty develop instructional content under the joint study.

Staffing is required to support the faculty from the inception of an idea, through the many courseware development steps (see Figure 2), to actual delivery in the classroom.[7] At present, Cal Poly's communications services department has (1) two full-time technicians supporting its audio-visual/television production unit and distance learning facility; (2) one full-time technician to install, upgrade, and maintain new hardware and software in the FMDC; and (3) several student assistants to do graphic design, digitization, editing, and authoring tasks. In addition, there is need for one full-time instructional designer to assist the faculty in developing the interactive multimedia courseware appropriate to their curriculum. This need should disappear as more faculty become familiar with the techniques and grow comfortable using the tools. These pioneers will become mentors and valuable campus resources as they begin to share their discoveries with colleagues.

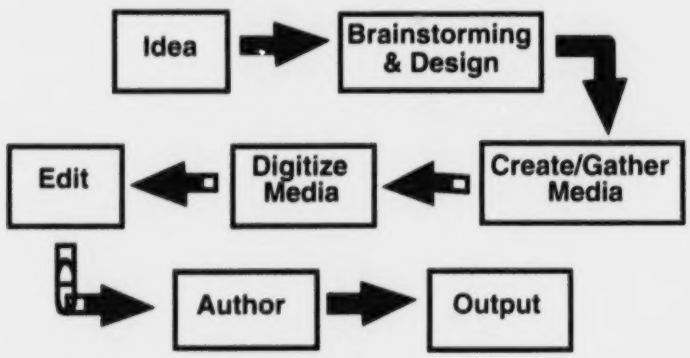


Figure 2: Courseware development

The Faculty Multimedia Development Center mentioned earlier is an important component of the support system. The center is equipped with both IBM and Apple authoring workstations and software tools, including image editors, video editors, and authoring packages. Other resources available to faculty include (1) scanners and digitizing stations to convert source materials from word processing, VHS tape, laserdisc, CD-ROM, illustrations, and artwork; (2) full video production facilities, including a videotaping studio; (3) hand-held video cameras for off-site work; (4) digital, video, and sound editing studios; and (5) in-house support for creating VHS tapes and CD-ROMs. These facilities were developed using existing audio-visual resources, combined with donated and discounted equipment.

The desire to use electronic technology in the classroom must, in the end, come from the faculty itself. To gauge faculty interest in this new technology, Academic Affairs encouraged faculty to submit proposals for release time and offered modest support to develop related projects this year. As it turned out, the campus was able to support only a small fraction of the expressed interest. Currently, faculty in nearly every discipline are involved in creating multimedia presentations for classroom instruction and professional meetings, and interest is steadily increasing. The campus is seeking matching funding, through various sources, to implement a more broad-based faculty training and development program, possibly in cooperation with other CSU campuses.

Reducing costs through partnerships

Cal Poly can only achieve its vision by forming partnerships in which the cost to operate and maintain the information technology infrastructure necessary to deliver education in the future may be partially offset by joint development projects with information technology vendors and other institutions and organizations. These projects must be mutually beneficial for both partners, and involve research, development, and testing of new technologies with potentially wide application to higher education beyond this campus.[8]

Over the years, the University has been successful in developing strong and lasting partnerships with many information technology vendors, including Hewlett-Packard, Pacific Bell, AT&T, SP Telecom, and IBM, to name a few. IBM has been a particularly strong ally in this regard, providing hardware, software, training, and support for key infrastructure projects supporting administrative and instructional computing. With their support, Cal Poly has taken its first steps towards becoming a virtual university. Other corporate partnerships include the following:

- With more than 250 Integrated Services Digital Network (ISDN) lines on campus, providing simultaneous access to telephone and network services, the University is working with Pacific Bell to extend ISDN service to faculty, staff, and student residences, including private residence halls, in the local community in 1994.
- The University is partnering with BellCore to implement SuperBook, an electronic document "browser" that can deliver library materials, journal abstracts, and other documents with text, graphics, and video to the desktop via the network. One major hurdle to address involves licensing and copyright protection of intellectual properties owned by the University, publishers, or faculty. Transactional monitoring and pricing techniques are being explored in a joint study between Cal Poly, Bellcore, Lawrence Livermore Lab, Chevron, and Pacific Bell.
- The University is participating in a joint study with The Robinson Group (TRG) and IBM to test using touch-screen kiosks linked to the University's student information system to allow students to check their own records for information about grades, account balances, current term registration, and other routine requests currently handled in person, by phone, or through the mail. Also under review are methods to allow students to directly update data such as address changes.
- Most University faculty use Macintosh or IBM-compatible computers to develop course content. Since the MegaServer currently supports only IBM-compatibles, Cal Poly and IBM began beta testing Macintosh support earlier this year to extend full-motion network access to Apple computers.
- Another partnership with IBM is enabling testing wireless network access. This technology will permit faculty to access the MegaServer from any classroom, using a transmitter attached

BEST COPY AVAILABLE

to a laptop computer. If viable, this could eliminate the need for specialized facilities, reduce costs, and greatly expand campus access.

Cal Poly continues to seek private and corporate grants and other external funding for related distance learning, multimedia, and telecommunications projects. A group of faculty has already submitted a proposal to a major national foundation interested in how this technology might be used to deliver a full-degree program to students at home. They are especially interested in the techniques faculty would develop to foster collegiality and shared group interaction between students and instructors and among the students themselves by using communication technology.

The University also recognizes the value of partnerships and collaboration with other education institutions:

- Cal Poly is working toward expanding network capability to other parts of California, through pilot projects with telecommunications vendors to develop and test high-speed, gigabit networks; has established distance learning partnerships with Bakersfield College and Cuesta Community College, to jointly develop and share course materials to facilitate instruction at both levels; is pursuing an ATM test link with CSU Hayward to allow the two campuses to share digitized course materials and interactive instruction; and is expanding access to K-12 schools, to provide college-level courses, including Advanced Placement, to high school students.
- The University's College of Engineering, along with the seven other universities in the National Science Foundation National Synthesis Coalition, are creating a National Engineering Educational Delivery System (NEEDS) that will not only advance the curriculum and enhance the classroom environment, but also promote faculty collaboration and give students direct access to a vast database.
- A major publisher has already shown considerable interest in the work of some Cal Poly faculty who are developing multimedia courseware. If local faculty don't develop their own materials, they can use courseware created by colleagues elsewhere and modified as needed for their classes. For example, Cal Poly and CSU Long Beach are jointly developing a distributed database of digital information (images, audio, full-motion video, and so forth) that will be able to accommodate potential contributions from faculty in any discipline and on any campus. Once developed, faculty on any CSU campus will be able to query the system by data type (audio, graphic) or subject, and retrieve files remotely for inclusion in a classroom presentation or courseware module.
- The CSU is exploring a partnership with the State University of New York (SUNY), the City University of New York (CUNY), and a private academic systems development firm to support faculty in creating mediated learning courseware in courses that specifically create barriers to students who would like to pursue science, mathematics, or engineering programs.

What's next?

Many faculty are burdened with older workstations incapable of supporting the full-motion video and other resources envisioned as part of the "virtual university." Over time these systems will be replaced, but it will take a concerted effort on the part of the colleges to ensure that faculty are equipped with the resources they need.

While almost anyone on campus with a computer and the proper connectivity can now participate in electronic mail and some other resources, the level of service is uneven across campus. With the growing interest in technology-mediated instruction, the IRM Policy and Planning Committee has recommended a new set of communications goals, which will mean much more sophisticated installations to all offices, classrooms, labs, and even the dorms. The network will become simply another campus utility, like the phone system. Higher bandwidth will allow faculty to take full advantage of the information resources.

For off-campus users, private information servers and other public utilities will put these more sophisticated communications tools in the hands of students and members of the public wishing to link up with the University system. Cal Poly is already working with local government and industry leaders to make San Luis Obispo an "electronic village," by extending the network into the community as quickly as possible. Internet access and local network services are already being offered on a limited scale, but to truly bring the benefits of the virtual university to the home will require the support and cooperation of local telecommunications

vendors.

We do not expect to achieve these goals all at once. Instead, we intend to proceed deliberately, while keeping abreast of changes in technology that may suggest new directions, and the developments in public and private communications ventures that will provide ubiquitous broadband networks. Still, we feel that we must begin proceeding now toward a networked instructional environment if we are to deliver the sort of education our students will need as we move into the next century.

Footnotes:

[1] For a more extensive discussion on the issue of learner productivity and higher education, see D. Bruce Johnstone, "Learning Productivity: A New Imperative for American Higher Education," *Studies in Public Higher Education* No. 3 (Albany, N.Y.: Office of the Chancellor, State University of New York, 1993), pp. 1-31.

[2] Norman Coombs, "Teaching in the Information Age," *EDUCOM Review*, March/April 1992, p. 30.

[3] Chen-Lin C. Kulik and James A. Kulik, "Effectiveness of Computer-Based Instruction: An Updated Analysis," *Computers in Human Behavior*, Vol. 7, Nos. 1-2 (1991): 75-94. See also W.D. Sawyer, "The Virtual Computer: A New Paradigm for Educational Computing," *Educational Technology*, January 1992, p. 21; and Loretta L. Jones and Stanley G. Smith, "Can Multimedia Instruction Meet Our Expectations?" *EDUCOM Review*, January/February 1992, pp. 39-43.

[4] See Richard Lanham, *The Electronic Word: Democracy, Technology, and the Arts* (University of Chicago Press, 1993) for a discussion on how "digitization of the arts radically democratizes them" (pp. 105-107).

[5] These plans are described in Cal Poly's *Campus Information Resources Plan: 1989-1994* (CSD-0369) and *Campus Information Resources Plan: 1990-1995* (CSD-0918). Both are available from the CAUSE Information Resources Library (orders@cause.colorado.edu or phone 303-939-0310).

[6] *Master Plan for Higher Education, A Dream Deferred: California's Waning Higher Education Opportunities*, California Postsecondary Education Commission Report 93, June 1993, p. 10; see also James Ogilvy, "Three Scenarios for Higher Education: The California Case," *Thought & Action: The National Education Association Higher Education Journal*, Vol. IX, No. 1 (Fall 1993): 25-67.

[7] The importance of centralized support is discussed in Fred Hofstetter, "Institutional Support for Improving Instruction with Multimedia," *EDUCOM Review*, January/February 1992, pp. 27-30.

[8] Arthur S. Gloster II and James L. Strom, "Building Strategic Partnerships with Industry," *Information Technology: Making It All Fit, Proceedings of the 1988 CAUSE National Conference* (Boulder, Colo.: CAUSE, 1989), pp. 263-268.

==========

Sidebar 1:

Cal Poly: Becoming an Electronic Campus

The University provides access to all major resources through its Fiber Distributed Data Interface (FDDI) backbone network that links thirty-nine core campus buildings and residence halls.

- The network serves more than 2,400 student residents on campus and provides connectivity to most of the University's 900 faculty and 1,200 staff.
- More than 13,000 of Cal Poly's 15,000 students have electronic mail accounts.
- More than one-third of the fall 1994 applications for admission were submitted in electronic form by incoming students.
- Online administrative systems provide timely access to student records, class schedules, financial aid, grades, and other information.
- Increased use of electronic mail, calendaring, online reporting and requisitioning, and tools such as Gopher and other online services has reduced costs and changed the way

departments and individuals communicate and request information.

==========

Sidebar 2:

The CSU's Project DELTA

The California Master Plan for Higher Education, initiated in 1960, calls for access by all eligible students to the three-tiered higher education system in California. For the California State University, this means that all high school students graduating in the top third of their class are eligible for admission. Given current economic conditions in the state, it is unlikely that the CSU system will be able to expand its physical facilities to meet the increased enrollment demand generated by the master plan. Instead, the system must meet that demand by offering new ways to deliver the required education to students both on- and off-campus.[fn]

The CSU Commission on Learning Resources and Instructional Technology (CLRIT) was created to investigate options for using electronic technology in education. Its first major initiative, Project DELTA (Direct Enhancement of Learning Through Technology Assistance and Alternatives), provided seed money for multi-campus projects designed to:

- improve instructional quality and effectiveness;
- increase student access to higher education, by making access more convenient; and
- promote greater productivity and accountability in the use of public funds.

CLRIT is also providing oversight and guidance in the development of systemwide library planning through "Knowledge and Information for the 21st Century," a strategic plan for CSU libraries being prepared by the CSU Council of Library Directors, and in telecommunications planning through "Leveraging the Future: The Telecommunications Plan for CSU," being developed by the CSU Academic Communications Network Committee.

==========

Sidebar 3:

Virtual University: Potential Applications

Delivery of education to students in classrooms at multiple CSU campuses:

- capturing unique faculty experts and special lecturers on video as a way to augment lectures/courses
- downloading information from multiple sources into a multimedia presentation in the classroom
- teaching low enrollment courses at multiple campuses using two-way video
- evaluating student teachers remotely in the classroom and communicating via electronic mail
- teaching remediation courses at CSU campuses remotely from community colleges
- conducting library/text searches online
- requesting assistance via e-mail with timely responses from faculty
- interaction among students and between students and faculty utilizing bulletin board or conferencing software

Delivery of education to non-traditional, off-campus students in their workplaces or homes:

- specialized training and retraining programs for industry
- professional licensing/certification courses
- adult education/enrichment programs
- continuing education or degree credit programs
- Advanced Placement courses to high school students

Streamlined administrative services to students:

- apply for admission, financial aid, housing, and so forth to one or more campuses using customized electronic forms
- transmit financial aid data to "Sallie Mae" and a third party for more timely evaluation and electronic fund transfers to students and campus
- analyze articulation requirements between schools, community colleges, and universities
- apply AACRAO Electronic Data Interchange (EDI) standards to build databases for capturing transcript/other data
- distribute test scores, grades, transcripts, coded memoranda, and other documents

==========

***Warren J. Baker**, President of California Polytechnic State University since 1979, is a leader in the implementation of academic computing systems. He chairs the California State University's Systemwide Commission for Learning Resources and Instructional Technology. Appointed in 1985 to the National Science Board (NSB), Dr. Baker has served on the NSB Executive Committee and chaired the Programs and Plans Committee for five years. In that capacity he conducted Board reviews of the National Supercomputing Centers and the NSFNET.*

***Arthur S. Gloster II** has been Vice President for Information Systems at California Polytechnic State University, San Luis Obispo, since 1986, overseeing campuswide academic and administrative computing and communications. With more than twenty-five years experience, he is regularly consulted by the public and private sector on information technology issues and management. He served on the CAUSE Recognition Committee for the past three years, and is a regular presenter at CAUSE and other national forums on using IT to meet higher education goals.*

UNIVERSITY OF FLORIDA

Productivity Tools: An Executive Insight and Evaluation

Presented by

Mr. John E. Poppell
University of Florida
Gainesville
Florida

How can we develop and deliver increasingly better systems applications at the same time we are experiencing an almost annual reduction in resources? Many institutions--large and small, public and private--are facing this dilemma today. The University of Florida, like most institutions of higher education, depends heavily on its administrative computing group to develop, deliver, and maintain economical and effective applications that enhance processes and optimize resources.

This nontechnical session will deal with the Universities' investment in productivity using Computer Aided Software Engineering (CASE) tools. More importantly, it provides an executive's evaluation of this investment. Can investing in CASE tools provide a return sufficient to warrant the cost? Is CASE the future of application development? This presentation on CASE tools, based on UF's two years of experience, will provide answers to these and other related questions. Productivity tools are not new, but the results they are bringing to the early users are!

Productivity Tools: An Executive Insight and Evaluation

The very mention of the word "productivity" in management circles today seems to invoke apathy and a callous disdain. The word productivity, like the word quality, is often overused, but still recognized as a key to survival in the business of higher education. What is true productivity? Can it be measured? Have real results been realized? Can these results be quantified? The University of Florida can now present a case for investing in productivity and can provide evidence of returns on that investment.

In what specific area within my institution will this investment be made?

The answer to this question is as follows. Information Systems (computer-based applications) is a large and critical component for managing and processing the University of Florida's operational responsibilities. This division handles the accounting system, the payroll system, receivable systems, and inventory systems. Yes, our dependence on computers and the applications developed to operate on those computers is both critical and absolute to our operational survival. Given this, and regardless of your institution's size or mission, there is an excellent chance that an institution that relies heavily upon some form of computer technology can enhance productivity.

How?

To establish a reference point, let's return to the late 1960's and early 1970's. Architectural and engineering professionals had just embarked on something called "CAD/CAM" - Computer Aided Design/Computer Aided Manufacturing. Simply stated, architects and engineers were in need of a systematic and standard approach to design and build aircraft carriers, bridges, skyscrapers, submarines, and other very large and complex structures.

The need for a "recipe" that would standardize each task, beginning to end, was the challenge. The use of CAD/CAM provided that recipe and has withstood the test of time. Essentially, the same process and system are employed today to design and build most structures. Refinements have occurred, but the foundation of a scientific approach to design and construction was forever formed.

So, what does this have to do with higher education and development of "application software"?

Higher education, not unlike the private sector, has traditionally approached software application development with little regard for a standardized approach. Software application development was more artful than scientific, typically representing the systems analyst's style and skill level, both in design and construction. For years, this type of approach has caused systems development to be costly, somewhat unpredictable, and, most times, late into production. This type of approach was often unpleasant and frustrating for the end users and, naturally, to the managers of the operational units who requested the new application. Not only does this artful

approach cause frustration towards new application development, it also contributes to very high maintenance demands. It is not uncommon to find 50% of a development department devoted to the maintenance of legacy systems.

Why is there such high maintenance demand?

Given there were few standardized processes from the beginning to the end of both design and construction of programs, software analysts were constantly searching for the secret designs used by their predecessors to develop the software. Given that the employee turnover in this profession is somewhat high, even good systems were found to be mystical after 5 or 6 years in operation.

What, then, has occurred that can now increase productivity in application system development?

The answer is "CASE" technology - Computer Aided System Engineering, the CAD/CAM of software design and construction, and otherwise known as "productivity tools." It is this technology and a systematic approach that can and will provide software application developers new gains and increased productivity. These CASE tools have been around since the late eighties in varying forms, styles, and complexities. Most tools deal specifically with design or program construction, and some even integrate both processes. Today, there are comprehensive tools for all needs, regardless of your institution's size or needs.

So, what's new?

What's new is the results that are coming in from departments that have introduced CASE tools and a systematic approach. Development departments early into this new approach are experiencing the return on this investment in productivity. End users are beginning to benefit from this new approach.

So, let's have it - what are the results?

- Better-designed systems
- Self-documenting systems
- Systems delivered on time
- End-user ownership from the beginning of the project
- Employee morale is up
- Maintenance is cut some 40-50% on new systems
- Productivity is up 10-40% on all projects

These results are real - no sales pitch - no wishful, hopeful theoretical promises. Productivity is up and the investment is now paying returns.

Can the results be quantified?

The answer to this question is, regretfully, yes and no! Yes, later in this paper I have provided some quantified data that depicts documented savings. Ironically, the better answer is no, because we at the University of Florida have essentially discontinued "old" customary practices in favor of the new CASE technology. So, exact comparisons are naturally impossible, but we have keen memories and our very experienced analysts and project managers know how it "used to be." It is from these comparisons that we draw our conclusions.

What are the quantified results?

Conventional vs. CASE Tool Development

	Task Hours Conventional	Task Hours CASE	% Savings
Design			
Data/DB2	180	126	+30%
Applications	384	441	-15%
Construction			
On-Line	1100	594	+46%
Batch	475	200	+58%
Testing			
On-Line	266	186	+30%
Batch	209	135	+35%
Documentation			
Internal	135	95	+30%
User	210	189	+10%
Maintenance			
On-Line	300	120	+60%
Batch	410	254	+38%

It is important to remember the "Conventional Method," given our current, complete use of "CASE," is now an estimate; however, we believe the estimate is very accurate.

What does the investment cost?

Tough question. The answer depends entirely upon the level of CASE technology you wish to employ. The University of Florida undertook a comprehensive project to employ CASE. We installed an IBM OS/2 LAN to develop applications for our campus main frame, an IBM ES 9000. Individual workstations (IBM PS models) were equipped with a complete CASE tool package. All development staff now use these CASE tools. This has become the standard. The average cost, including hardware and software, is approximately $15-20,000 per workstation. This cost is exclusive of training, which is also necessary if the CASE tools are to be effective.

What are some of the tools?

Tools purchased by the University of Florida

Product Name and/or Company	Application Functions Supported	Cost*
Excelerator II for Client/Server V2.0 Intersolv	A planning, analysis, and design toolset for developers of IS applications. Used to create detailed descriptions of applications components, diagrams that show the relationships among these components, and GUI windows or character screens that represent the entry points to the application.	9,500
APS for Client/Server V3.0 Intersolv	A full-function application generator that automates the development and redevelopment of MIS applications to support business. Used to build applications for a variety of production environments using high-level specifications and little or no manual coding.	9,500
Micro Focus COBOL	Software that provides everything needed to develop COBOL applications to run under DOS, Windows, and OS/2 on IBM compatible computers.	
Micro Focus Toolset	An add on product to Micro Focus COBOL development systems to provide a powerful application system.	
Micro Focus COBOL Workbench	A set of software tools to help develop COBOL applications using Micro Focus COBOL for IBM and IBM Compatible computers.	2,250**
Micro Focus CICS	A complete transaction processing development environment targeting development of CICS/ESA COBOL applications. Provides a flexible integrated toolset to enable development of cooperative and distributed processing systems.	1,250
Bachman/DBA	A software tool for building/analyzing, and modifying designs for IBM's DB2. It helps information professional to quickly build and optimize new DB2 designs, and allows them to easily examine and modify existing designs.	10,000
Bachman/Analyst	Provides an integrated set of modeling features that enables developers to build, analyze, and modify models representing complex information systems; and a way to develop and maintain information systems through data, process, and logic modeling.	25,000

* Does not consider educational discounts.
** Cost includes Micro Focus COBOL and Micro Focus Toolset

Other CASE Tools

Product Name and/or Company	Application Functions Supported
IEF for Client Server Texas Instruments	Full Life Cycle Support
AD/Method Structured Solutions	Customizable Automated Methodology
Application Development Workbench (ADW) Knowledge Ware, Inc.	Full Life Cycle Support
Automated Testing Facility (ATF) Softbridge, Inc.	Automated capture, playback, scripting approach to testing
C Softbench Hewlett-Packard	Construction Tool (C development)
CA-Realia Workbench Computer Associates	Construction Environment (COBOL)
Data Dictionary/Solution Brownstone Solutions	Corporate-wide design repository
DB Analyzer Info Systems Group	Relational Database design analyzer
Foundation Anderson Consulting	Methodology and Construction tools for corporate wide systems
Natural & Natural Construct Software AG	Code Generator and 4GL
PowerBuilder Desktop Powersoft Corp.	Construction Environment for OO
System Architect Popkin Software & Systems	Methodology Design Tool and Generators
Visual Age IBM	Object Oriented Development Environment

Can it work for my institution?

The answer is yes. In fact, most experts agree CASE will do to software development what CAD/CAM did for engineering. It will become the accepted standard for development. It can and will work in any software development environment: academic, administrative, etc.

Should you wait?

This answer is easy. There are no short-cuts in the employment of CASE. The training and learning curve is constant. Waiting will only delay the inevitable and prohibit productivity gains.

There are numerous CASE tools available, with many new vendors entering the market place. Selecting the tools for your institution is not much different than most procurement decisions. Seek the advice of your information systems colleagues or visit institutions that have some experience. Private-sector systems departments are likewise entering this new technology. User

groups are forming and can help guide your institution. Productivity gains can be achieved. At the University of Florida, the results are in!

What observations have been made?

Overall, all facets of the application development process, except one, have experienced productivity gains. The one exception, as expected, is the JAD, Joint Application Development. It has been our experience that involving the end users early in the planning and design stages was difficult. Users would typically describe a problem and possibly outline a perceived solution, and wait for the analysts to bring structure and organization to the solution. The implementation of JAD has forced users to participate in the design process and to make critical design decisions at the appropriate times. This added time is very well spent, is welcomed by the development staff, and is a significant contribution to the saving of time in the latter stages of the project.

What is our overall assessment?

The use of CASE, or productivity, tools has been good for the University of Florida. We are realizing significant savings. The investment is paying large returns in real time savings, higher employee moral, and better client satisfaction! We believe it was a wise investment, one that will be long lasting; much like CAD/CAM was for architects and engineers.

DESIGNING AND IMPLEMENTING THE CUNY OPEN SYSTEMS CENTER

Richard F. Rothbard
Vice Chancellor for Budget, Finance and
Information Services
The City University of New York

Michael Ribaudo
University Dean for Instructional Technology
The City University of New York

Colette Wagner
Director of Education, Training and Staff Development
The City University of New York

Professor Michael Kress
Computer Science Department
The College of Staten Island/CUNY

James Murtha
University Dean for Computer Information Services
The City University of New York

DESIGNING AND IMPLEMENTING THE CUNY OPEN SYSTEMS CENTER

Richard F. Rothbard
Vice Chancellor for Budget, Finance and
Information Services
The City University of New York

Michael Ribaudo
University Dean for Instructional Technology
The City University of New York

Colette Wagner
Director of Education, Training and Staff Development
The City University of New York

Professor Michael Kress
Computer Science Department
The College of Staten Island/CUNY

James Murtha
University Dean for Computer Information Services
The City University of New York

I. THE ROLE OF REORGANIZATION

by Richard F. Rothbard

Abstract: The creation of the CUNY Open Systems Center as an outcome of a reorganization of the central computing enterprise of a large urban university system will be introduced.

In July of 1993, the CUNY Office of Budget, Finance, and Computing embarked upon an ambitious reorganization plan. The plan was the result of a thorough examination, conducted by colleges and representatives of the central administration, of the needs of the University and its colleges in the areas of computing, telecommunications, and related technologies, and the resultant call for us to exercise leadership in planning for and implementing technological solutions, where appropriate, for the many academic and administrative challenges facing CUNY.

Far from representing a mere name change to the Office of Budget, Finance and Information Services, CUNY's central computing enterprise has undergone a top to bottom reorientation to serve better the needs of the colleges and to position better the University to make the most effective

use of current and emerging technologies in the service of higher education. And the emphasis is very deliberately on Information Services and not Systems, in recognition of the fact that systems are merely tools that may be helpful in achieving an objective, not the objective itself. Rather, our goal is to provide services in the new information age, services that all of us are either required to perform by internal and external mandates, or want to provide by virtue of our shared notions of how to improve life for our students, faculty and staff.

One of the first major outcomes of the reorganization was the Fall 1993 inauguration of the Open Systems Center, a high-end research and training facility located in the Computer Information Services offices at the University's central hub that is designed to serve as a testbed for the application of new technology to problems encountered by the University's professional staff in teaching, research and administration. This morning, Michael Ribaudo, University Dean for Instructional Technology and Industry & Government Partnerships will offer a look at how the creation of the Center emerged from the reorganization and how it took technological shape. Colette Wagner, University Director of Education and Training, will report on the diverse instructional activities currently under development at the Open System Center, Mike Kress, will then speak to you briefly about a number of advanced technology projects that he and his graduate students have been working on and Jim Murtha, University Dean for Comuter Information Services, will address issues regarding sustaining the Open Systems Center effort. Since I'm sure with the diversity of interest represented in this group our prepared remarks might not address every aspect of the project, we'll try to leave ten to fifteen minutes at the end of the session for discussion and questions -- as well as to clear up as much as we can the inevitable confusion that may be plaguing you at the time.

II. BUILDING A PHYSICAL CENTER AND FORMING A VIRTUAL TEAM

by Michael Ribaudo

Abstract: The basic philosophy of the Open Systems Center and a chronicle of its development and operation will be presented. Issues to be covered include: relationship of the initiative to the reorganization of the CUNY Information Services enterprise, installed equipment base and overall strategy.

The Open System Center we are talking to you about here today is actually a small physical embodiment of a larger philosophical construct that has guided the progress of

central computing at the City University of New York since the computing and telecommunications area for the Central Office was reorganized into its current structure in July of 1993. Prior to that, much of the thinking which governed the delivery of central computing services could be characterized as largely "Mainframe-Centric". The environment was more like the traditional 'glass house' computer center, with few staff having had any exposure to UNIX based workstations; the orientation of the networking infrastructure being almost totally limited to an IBM/327x/SNA view of the world. In fact, at the time of the reorganization only one or two of the forty or so full-time staff members had anything on their desktops other than a dumb terminal.

In the year and a half or so since the reorganization things have changed considerably. All staff members now have either Intel 486-based or Macintosh machines on their desks, every workstation is connected to a ubiquitously deployed high-speed ethernet network, and we've introduced a suite of servers running almost half a dozen varieties of the UNIX operating system. We haven't as yet thrown away any of our mainframes, but the work those mainframes do now is somewhat different from what they've done in the past. Our central mainframes are now doing what smaller campus based mainframes did for the last twenty or so years at CUNY. By aggressively engaging in a system-wide program of mainframe consolidation, we have been able to afford considerable savings to the CUNY colleges which in increasing number are choosing to run their own locally based administrative mainframe systems on the central processors. By the end of this academic year, over one third of the CUNY colleges will be running their administrative systems"--"for the most part, their student registration systems"--"at the University's central computing facility on West 57th Street in midtown Manhattan.

The shift in central mainframe workload from a primarily academic orientation to a more administrative one provided the initial impetus for creating the Open Systems Center. In order to free up the mainframe cycles and DASD cylinders to accommodate the college administrative systems, we needed to provide alternative and perhaps more appropriate platforms for users looking to port their applications elsewhere. While our utilization studies were telling us that instructional use of the mainframe by students had trailed off considerably over the years, faculty research usage remained high. The time had in fact come for folks like our social science researchers who over the years had become whetted to running SAS or SPSS on the mainframe under MVS to look at the approaching millennium.

Another compelling reason for establishing the Center was our desire to provide faculty interested in developing multimedia courseware applications with a central site where those skills could be developed and nurtured. The university had

already obtained site licenses for a number of high end authoring languages and tools and a number of the applications written internally for instructional purposes have won critical national acclaim.

A third rationale was to set up a model site for local area network strategies that the colleges could look to for guidance as they seek more and more to implement campus-based connectivity solutions which include mechanisms for delivering multiple media to the desktop and high speed connections to the Internet.

We wanted a Center that embodied the then emerging philosophy the 'network' was becoming the computer and that in order for that network to accommodate the variety of vendor hardware platforms and operating systems different constituencies would require, we would need to build an open network capable of carrying a variety of network protocols. To that end we have built a heterogeneous and versatile computing environment in a three room 80,000 square foot setting. It currently houses 24 high-end multimedia computer systems and six UNIX work stations and a variety of peripheral devices such as printers and scanners in a state-of-the-art networked environment. The hardware includes Apple Macintosh 840AVs and Power Macs; DEC Alphas; IBM 486-Value Points, Pentiums, and RS-6000's; and Sun Sparcstations all connected in a Novell network capable of carrying Netware, Apple Talk, and TCP/IP over level 5 10-base T unshielded twisted pair wire and gatewayed to the Internet through a Cisco router which is scheduled to be upgraded from T1 to T3 speed within the next few months.

Staffing the Open Systems Center has been a challenge. From the outset, it was clear that no one staff member possesses the full range of skills required for its effective operation. As a consequence, a virtual team has been constructed, pairing staff with requisite skills in project-based activity across traditional organizational boundaries. While successful to some degree, this virtual team approach has created ambiguous situations to which some staff have been unable to adjust. This places divisional leadership in the ironic position of arguing for new staff lines not based on the need for more staff members but based on the argument that existing staff do not possess the requisite skills set to achieve newly promulgated organizational goals. On various projects, campus staff are added to the mix as well. We continue to struggle with this thorny issue through a variety of strategies, not the least of which is our education and training program about which you will hear in a moment.

Finally, a word about start-up funding. Creative partnerships with hardware vendors enabled us to maximize the available budget with the result that a fairly impressive installed

equipment base was available from the outset. In our second year of operation, the emphasis is on equipment upgrade and building the software tools available to Open Systems Center users. We are constantly seeking ways in which to finance these activities.

III. TRAINING FOR NEW TECHNOLOGIES

by Colette Wagner

Abstract: The Open Systems Center provides an umbrella structure for training faculty and computing staffs across the University. Its workshop "curriculum" spans interests from introduction to the World Wide Web and HTML document development, to an introduction to Unix for mainframe programmers, to training in the use of Universal Algorithms Schedule 25 (a facilities management program). In addition, the University's Multimedia Courseware Development Initiative is funded under the aegis of the Open Systems Center. The rationale for linking these training experiences under the Open Systems Center banner is explored.

In my capacity as Director of Education and Training for the Office of Instructional Technology, I am the Program Coordinator of the Open Systems Center and it has been my responsibility to develop the activities and agenda of the Center. A key element of that programming activity is the schedule of training workshops for faculty and computing staffs that take place under the umbrella of the Open Systems Center. A key concept in the design of the Open Systems Center's programs is that of the virtual team. From the outset, I have billed all training opportunities provided to CUNY faculty and computing staffs that emanate from the Office of Education, Training and Staff Development as programs of the Open Systems Center, whether they were physically located at the Center or remotely located at campus sites. The method to this particular madness is quite simple. The current space allocated to the Open Systems Center does not include a classroom/conference room environment that will accommodate groups over 25 and I need to locate larger programs at campus sites. In addition, I am slowly starting to build a cooperative network of University-wide training opportunities that will expand based on the deployment of campus-based Open Systems Center facilities at CUNY senior colleges about which Jim Murtha will speak at the end of our session. Finally, with the blurring of distinctions between instructional and administrative computing functions within the overall CUNY computing organization, the Open Systems Center provides a comfortable environment for the collapse of these traditional distinctions in the technology organization.

It might interest you to know that the entire full-time

complement of my training staff is myself and an assistant who publicizes workshops and handles registration, etc. In the period July 1993 through Fall 1994, we offered a total of 72 events, representing a total of 1300 participants and 1600 applicants. As you can well imagine from these statistics, the "faculty" of the Open Systems Center's training program are themselves a virtual team representing the best of CUNY's instructional and technology experts. This fall, four CUNY faculty members"--"Michael Fitzgerald (Philosophy, Medgar Evers), Michael Kress (Computer Science, College of Staten Island), Anthony Picciano (Curriculum and Teaching, Hunter College) and Dean Savage (Sociology, Queens College)"--"are serving as Visiting Faculty Fellows. In addition to 18 formal courses for faculty and instructional staff that they are teaching, one of the Open Systems Center's Visiting Faculty, Anthony Picciano, will be serving as a mentor to CUNY colleagues who are novice multimedia developers. This committee of Visiting Faculty Fellows provide the primary input in the design of new course offerings for the Spring 1995 semester and beyond.

Programming for faculty in the Open Systems Center is strongly tied to the University's instructional technology agenda. One of the Center's main objectives is to provide an experimental environment for prototyping instructional software that can be used in the real world of teaching, learning and research at CUNY. In particular, close ties exist between the Office of Instructional Technology's Multimedia Courseware Development Initiative (which has funded the development of approximately 30 multimedia projects by CUNY faculty since its inception in 1990-91) and the faculty workshops that are offered.

In its first year of operation, the Open Systems Center quickly became the locus of the Office of Instructional Technology's Faculty and Staff training workshops. In addition to the scheduling advantage afforded by a training center dedicated to faculty and technical staff, the specialized equipment and high-speed network connections that were designed into the complex enabled cutting edge programming from the outset. In Spring 1994, the Open Systems Center provided the University's first workshops on navigating and authoring documents for the World Wide Web and, as a consequence, this work spurred the development of CUNY's own home page. This semester, campus home pages and individual faculty home pages are all the rage. Additionally, short courses such as *Authorware Professional, Introduction to Data Analysis Using SAS for Windows, Preparing the Electronic Lecture,* and *QStats and QData* (statistical management programs developed by the Queens College Sociology Department and distributed free under terms of an NSF grant) were featured offerings of the Open Systems Center schedule. Small groups of faculty working on art and technology multimedia projects, and

foreign language faculty also used the Open Systems Center to plan events or review new instructional software developments. As the University's videoconferencing/distance learning technology project unfolds over Spring 1995, the Open Systems Center will become a locus for faculty experimentation with the new technology. Finally, as faculty requests for support in identifying appropriate instructional technology materials come into the Office of Education and Training, the Open Systems Center is used as a clearinghouse and a program springboard. Social Science faculty and Foreign Language faculty will be working on conference events in their respective disciplines throughout Spring 1995.

On the front of advancing technical skills of computing staffs across the University, the Open Systems Center strategy has been somewhat different. In start-up mode, training for computing staffs has been offered on a limited basis, with few large scale events, many vendor-sponsored briefings and a number of workshops limited to specifically targeted training audiences. For example, training in CA-IDMS, which is the basis of the University's Student Information Management System, is limited to those schools who are either already participating in the program or are scheduled to migrate to the SIMS systems in the near future. In the area of client/server and Unix training, gradual steps have been taken. Central Office computing staff and users of the *Schedule 25* room scheduling application have been provided with a series of in-house seminars on survival in these environments to enable them to become acclimatized to these new environments and roll-out new applications. At this point, we are in the process of developing more specific training strategies that will identify central office staff for higher level training and that will enable colleges to participate in the same kinds of training at the lowest possible cost. The same approach has been taken in the area of Novell Netware administration. With the installation of a large number of local area networks in the Central Office, we have had the opportunity to assess various Novell training providers while addressing immediate organizational needs. Our long-term goal is to use this information to enable the successful negotiation of a University-wide training contract that will again provide colleges with lowest possible costs for upgrading staff skills in this crucial area.

Finally, on the issue of long-range planning for the training programs that are offered under the umbrella of the Open Systems Center. As a result of the reorganization of Computer Information Services, all CUNY colleges are currently engaged in the process of articulating their own technology mission and program statements. This process involves consultation with the Central Office and one of the areas covered by the activity is technology training needs. It is anticipated that the long-term agenda of our Open Systems Center training program will be forged by this

activity and that it will be further affected by alliances with faculty, with students, with CUNY computing staffs, and with the strategic partners identified by Dean Murtha in his presentation.

IV. PARTNERING FOR INSTRUCTIONAL ADVANCEMENT

by Professor Michael Kress

Abstract: The relationship between a large urban University's centrally located, high-end R&D technology center and a Computer Science department at one of its remote senior colleges will be explored. Student and faculty projects in multimedia development, video-editing and scientific visualization conducted using the Open Systems Center facilities will be discussed. Results of a student-taught video-editing workshop for CUNY faculty will be reported.

There were several important features to consider in developing a mutually beneficial relationship between the Open System Center facilities and a CUNY senior college, located a two hour commute away. It was important that they be able to provide each other with valuable resources and that they develop effective communications. This meant using appropriate file transfer techniques to share interesting applications and to disseminate information. It was also necessary to have a strong commitment to success on the part of both parties.

The Open System Center provided the state-of-the-art computer environment (hardware, software, and network) and funds for the students. Both the Staten Island students and faculty, aggressive in their use of new technologies, provided expertise in using and testing the equipment, selecting appropriate software, testing file transfer techniques and network performance, developing applications, and teaching workshops at the Center. The collective expertise and knowledge of evolving technologies contributed by the students was drawn from a wide network of users located throughout the World via bulletin board postings and user group events and meetings in the New York metropolitan area. The faculty expertise in research and development was essential to identifying critical issues and the detailed focus areas pertinent to cutting edge technologies.

Effective communication throughout the project was accomplished through e-mail, fax machines, telephone, voice mail, and CU-SeeMe video conferencing software augmented with conferencing speaker phones. The use of the videoconferencing facility significantly increased the quality of interaction, especially for groups, but it required a scheduling and set-up component to insure that the teleconferencing studios were

available at each site and the technical links for the connection established in advance. Concise, brief meetings were held as part of bi-weekly testing and software installation and upgrade sessions at the Center. The daily activity of evaluating and using evolving software and hardware was done at the remote site where one computer station of each of the three platforms supported in the Center was available.

Overcoming the distance constraint between the College of Staten Island and the Open System Center presented a special challenge for file transfer. One of the essential aspects of developing multimedia software effectively is transporting large digital video files from site to site. Various methods of file transfer were considered and tested. At first, we thought that FTP file transfer over the CUNYNet wide area network, would fill our needs. However, after hours of waiting for the transfer of a test file, we realized that a careful evaluation of the network performance at various times of the day was required to understand the feasibility of this method.

After empirical tests of network performance and time calculations based on observed transfer rates, it became clear that even unattended overnight transfers were not always practical. Other methods considered included: portable PC hard disk and Lap-Link computer to computer transfer, 150 MegaByte (MB) transportable removable hard disks, One GigaByte (GB) external SCSI hard disk, read/write optical drives, and write once/read many (WORM) CD's.

Ultimately, we concluded that there was no single way best suited to all circumstances. Network transfer could provide overnight delivery of files but was risky given the possibility of network failure. The use of 150 MB removable disks offered a number of significant advantages. It involved mature, standard technology at a low cost; it was easy to use on all platforms and featured archival backup. It was easy for the user to transport or could be sent by "sneaker net" or "snail mail". However, the transfer rate was limiting for motion video playback directly from the drive. The One GigaByte external drive offered speed and large capacity storage but at a higher cost. It also required (for the most part) that the user him or herself carry the 8 to 10 pound drive from site to site. The optical and WORM technologies as advertised offered cost-effective price per megabyte of storage. However, the startup cost and rapidly changing proprietary formats caused a "let's wait and see" opinion on their use in practice, especially since 20 different Colleges would ultimately be using the Open System Center.

The challenge presented in developing and testing software and hardware integration in the Center typically becomes clear immediately after the first interaction with technical

support as the features and behavior of the technology is advertised but rarely known by the software support group or developers themselves. The result is the need for a methodical step-by-step, hands-on testing and evaluation of each component in the system to identify the "features", limitations, and "work arounds" required to harness its power. Bulletin board listings and user group support are invaluable in this phase of the development.

The applications developed by the CSI group used multimedia technology for a variety of teaching applications from Scientific Visualization to American Sign Language. The common component of the applications was the use of "home grown" digital video in a highly interactive multimedia program. The activities of each project included shooting video footage, digitizing and editing motion video, writing digital video playback scripts, and developing and testing programs with content experts. Many digital video-capture and playback-boards as well as editing software packages and configurations were evaluated. They ranged in price from $400 to $2700. For our purposes, the inexpensive consumer boards ranked highest in overall value. Three presentation software packages -- *ToolBook, Authorware Professional,* and *Visual Basic* -- were used for different projects. The following Staten Island projects were initiated as part of the Center's development effort: *Applications and Techniques of Scientific Visualization* -- a multimedia program for teaching and learning visualization techniques in science and engineering; *Bon Jour* -- a multimedia program for learning conversational French; *The Magic Rabbit* -- a computer aided educational (CAE) program for teaching English tenses, aimed at children whose first language is American Sign Language (ASL); *ASL Dictionary and Tutorial Program* -- a multimedia program for learning ASL; *Colors and Shapes* -- a CAE program for teaching autistic children; *What Is Multimedia* -- a descriptive program for demonstrating multimedia techniques for CAE programs; *A Multisensory Calculus Program for Visually Impaired Students* -- a student controlled program using audio-tactile material for learning Calculus for blind and visually impaired students.

One of the exciting aspects of learning and using cutting-edge multimedia hardware and software is ascertaining and disseminating information. The classical sources of information, including library references materials, are of limited value. By the time printed paper makes its way to the library shelf, the hardware and software discussed are often obsolete and the information of little use. Magazines, trade shows, bulletin boards, and user group networks are the essential sources of information. The classical professor with years of theory and a firm mathematical foundation is no longer the renowned expert in solving the details of contemporary development and integration. The students are the experts and become the teachers in the use of the

technology! Fortunately for me, professors retain a role as content specialists. Some are also helpful in presenting the student's material to workshop participants. Following the clear gradient of contemporary information flow, a series of faculty and staff workshops taught by students and faculty were held at the Center. The workshops were: Digital Video Editing, Survival in a UNIX Environment, UNIX Script Programming, Graphical User Interface (GUI) Programming, Client/Server Computing, Performance Evaluation and Optimization in A Client/Server Environment. All but the last two were taught by a student-faculty team. Each contained a significant hands-on component with more than 85% of the workshop spent using computers. For the most part, the workshops were at least 1/2 day in length. The projects and handouts provided the workshop participants with practical applications for developing operational skills. Participant surveys indicated an overall favorable evaluation.

V. STRATEGIES FOR CONTINUED DEVELOPMENT

by James Murtha

Abstract: The continued development of the CUNY Open Systems Center and the special projects and strategic alliances that will drive its future agenda are explored.

Now that my colleagues have described how the CUNY Open Systems Center was initiated and have detailed the range of its current activities, I'd like to give some idea of how we intend to sustain the development of the Open Systems Center. Special projects and strategic alliances will be the key to the future of the Open Systems Center.

Building the University's technical infrastructure is an announced goal of the Office of Budget, Finance and Information Systems. With a $3 million capital allocation for an Educational Technology Initiative from New York State in fiscal 1995, CUNY has been able to offer one of two possible technology programs to each of its nine senior colleges. The University's Open Systems Center will figure heavily in the development of both program options. The first option is the establishment of a campus-based Open Systems Center facility which will emulate the design and philosophy of its central parent. The campus-based center will support instructional and research development and testing and will maintain close connections to the central facility. Cooperative projects between and among the various Open Systems Centers will emerge as the centers are installed and program activities are initiated.

The second option available to senior colleges under the Educational Technology Initiative is the choice of becoming a

remote site in the University's emerging videoconferencing network. Videoconferencing/distance learning technology has been a research pursuit at CUNY for some time. Switched wideband trials with NYNEX and Ameritech on instructional projects, use of the University's proprietary T1 network to sustain PictureTell installations for administrative purposes, and monitoring the progress of desktop videoconferencing programs such as CU-SeeMe are examples of the range of activities in this field. At present, we are anticipating creation of a University hub at 57th Street that will be connected via a video-enhanced CUNYNet (i.e., the University's proprietary T1 network) and that will be linked initially to five similarly equipped remote senior college sites. The Open Systems Center will play a formative role in the development of this project as it is slated to be the site of the first connection in the network between 57th Street and City College.

In addition to the videoconferencing effort, the future of Open Systems Center will continue to be formed by the technology agenda of the University. For example, research initiated at the Open Systems Center on the development of the World Wide Web and subsequent training in navigation of the Web and authoring HTML documents lead to creative thinking about the ways in which CUNY could participate in the burgeoning international development of digital resource collections on the Internet. A strategic alliance among the CUNY Office of Library Systems, The New York Academy of Medicine, the New York Metropolitan Reference and Research Library Agency (METRO) and the New York Public Library has resulted in a $275,000 award from the U.S. Department of Commerce's National Telecommunications and Information Administration to support the design, construction and demonstration of an electronic Consumer Health Information Network. The project will establish user-friendly microcomputer access at libraries, colleges and hospitals, allowing users to navigate among a wide range of databases, including the Breast Cancer Information Clearinghouse, AIDS Treatment News, Cancernet, Oncolink, Lymenet and the New York State Department of Health's gopher service, among others. Several services will be available in Spanish as well. The network will build on the existing infrastructure available at CUNY, and in order to ensure capacity for growth and interconnectivity, it will employ standard Internet protocols, hardware and software. As the Consumer Health Information Network progresses, the Open Systems Center will continue to play a role its development -- as a training and testing site.

Faculty activity and research interest will also drive the agenda of the Open Systems Center. Recently, CUNY received a grant of approximately $25,000 from the United States Information Agency to maintain and help develop a new gopher specifically tailored to teachers and teacher trainers in

English as a Second or Foreign Language working at locations worldwide due primarily to the activity of CUNY Basic Skills faculty who pioneered listservs on this subject using the CUNY mainframe as a resource over the years. Through the USIA gopher, which is called TES/FL, they will be able to obtain, at no cost, a wide range of pedagogical documents, many produced by the English Language Programs Division of the USIA, as well as lists of Binational Centers, and announcements of international conferences, fellowships, and employment opportunities. A specific reason for USIA's support for CUNY as the TES/FL gopher site is the University's prior creation of TESL-L, the listserv list, or electronic mail discussion group, for teachers of English as a second language. The TESL-L listserv membership includes more than four thousand teachers in 73 countries, making it one of the largest interactive listserve forums on the Internet. The Open Systems Center will continue to play a role in the development of this resource as we explore alternatives to mainframe-based listservices, etc.

We are also pursuing strategic partnerships that will further the development of the Open Systems Center. These partnerships can take various cooperative forms. Our latest venture is the establishment of a CUNY/New York Software Industry Association Internship Program under the aegis of the Open Systems Center. With economic development funding from New York State, this spring will see the placement of 50 funded interns in software companies in the greater metropolitan area. One of the anticipated outcomes of this university-industry partnership is the development of specialized training workshops to be housed at the Open Systems Center that will forge greater cooperation between education and industry to the benefit of both parties.

As a symbol of our reorganization and the reinvigoration of computing and technology at CUNY, we see the continued development of the Open Systems Center as paramount. Under its aegis, we will aggressively pursue all opportunities to research and evaluate the ways in which technology can improve teaching and support learning and research at CUNY, and afford us the ability to provide student services more effectively and efficiently.

Taking the Mystery Out of Document Imaging
Metropolitan State College of Denver

E. Leon Daniel
Associate Vice President for Information Technology

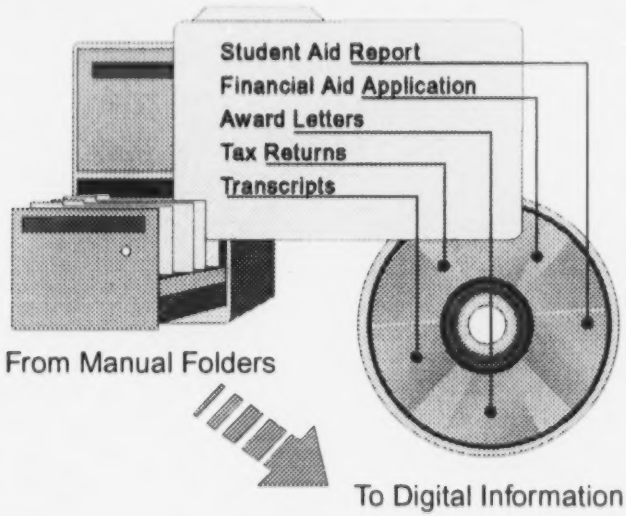
Steve Franzkowiak
Information Technology - Applications Services Manager

Abstract

This past year, Metropolitan State College of Denver implemented a Windows based archival document imaging system for the Financial Aid Office. The environment consists of twenty-eight IBM 486 image viewing stations (17" monitors), two Pentax 10 page/minute scanners, and a single HP LaserJet IIIsi printer. The system is designed to manage an unlimited number of student documents, and has an average image retrieval time of under 4 seconds to the workstation.

This application was developed over a six month period by a team of four individuals, using PowerSoft's PowerBuilder client/server development tool, in conjunction with Microsoft SQL Server. Developing the system internally provided a system that is closely integrated into the existing Banyan Vines network, and resulted in a cost savings of $140,000, as compared to the proposed vendor solution.

This presentation addresses many of the issues MSCD faced in making the decision to build the system as opposed to purchasing a package. In particular, issues relating to the use of new Windows based imaging technologies and application development tools will be presented.



Taking the Mystery Out of Document Imaging

Introduction

Each year the Office of Financial Aid at Metropolitan State College of Denver (MSCD) processes over 12,000 student applications for financial aid. This amounts to approximately 150,000 pieces of paper requiring extensive manual processing and filing. Rummaging through the reams of file folders to retrieve a student's document was time consuming and very inefficient, not to mention the space needed for storage. No doubt a tremendous amount of staff time is consumed in the never ending quagmire of paper, paper, paper. In an effort to address the document management and processing problems, the Director of Financial Aid sought Information Technology's help in assisting them to find an automated solution.

A systems analyst was assigned the tasks of evaluating Financial Aid's workflow and defining system requirements. The original project objective was to develop a Request for Proposal in order to procure a vendor package solution. This process took 6 months to complete and resulted in a detailed document of system requirements and a RFP. In parallel with this process, numerous vendor systems where evaluated and priced, so as to understand their computing platforms and potential costs. Imaging systems from IBM, DEC, Canon, and other vendors were considered.

As we learned, architectures for imaging systems fall into primarily three categories; 1) host or mainframe based, 2) networked (client/server), or 3) stand alone systems. Having recently completed the implementation of a campus-wide Banyan Vines network, a client/server solution was preferred. Exploiting the desktop had been an objective of MSCD's Information Technology strategic plan for the campus. The Financial Aid office was already targeted to receive new 486 PCs to replace their old 286 & 386 systems. These systems provided the hardware necessary to drive an imaging system. In addition, the existing fiber optic backbone provided enough bandwidth to transport images through the network.

After reviewing the major offerings, only one vendor met the bulk of our requirements. The cost of this solution was approximately $240,000 excluding the cost of user workstations, which exceeded available funding and forced us to reconsider other options. With the existing hardware and network infrastructure already in place we turned our attention to the possibility of developing the application in-house. Not having experience with imaging technologies, the development staff had numerous questions.

How do we capture, display, and store document images?
How do we tie the images to database information?
Must we retain the original documents or can they be destroyed?
What considerations are there to image compression and decompression?
How do we integrate optical disk technologies into our network environment?
How do we interface to the scanners from our application programs?
What database should we use?

How do we structure document profiles and catalog indexes?
Should we incorporate workflow capabilities?
How do we transport and view images through the mail system?
We've had no Windows development experience, what's the learning curve?

These questions and other concerns needed to be resolved before any decision could be made. One by one each issue was evaluated and technical concerns resolved. It soon became clear that it was indeed possible to undertake the development of the project. The decision to proceed with an in-house solution was made by the Associate Vice President for Information Technology and supported by the Vice President for Administration and Finance.

The Solution

Once the decision was made to develop an imaging based application, a focus group of four individuals was formed. The project team consisted of a Project Manager, Lead Analyst, and two Programmers. This group was relieved of day to day production responsibilities to focus on the project. Work requests from other areas were reprioritized to lessen the impact on remaining staff. The project was initiated in May of 1993 and the application was completed by the end of August 1993. The system was implemented in September 1993.

Project Time Frames

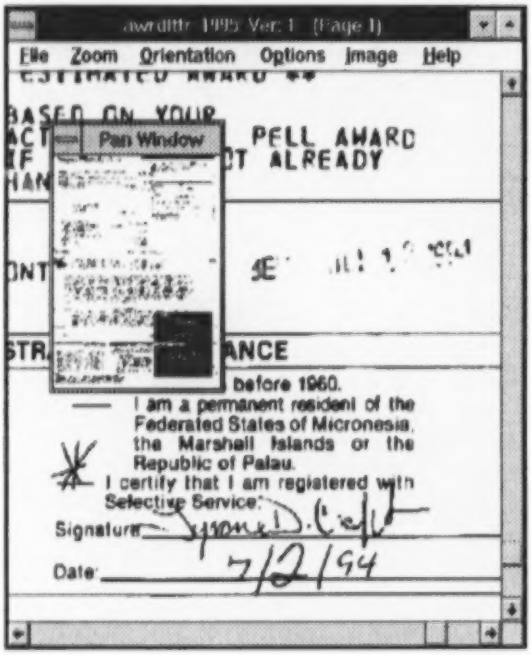
Develop RFP & System Specifications	6 Months
Analysis & Design	3 Months
Programming & Testing	2.5 Months
Installation	1 Month

Working with the Financial Aid department, the team finalized the application design and developed program specifications. The following list serves as an example of the types of requirements that were identified, and are in no particular order of importance or priority. A detailed listing of the requirements document and RFP can be obtained through the CAUSE document library.

- Must be a Microsoft Windows's compliant application.
- Retain catalog and images for at least 7 years (current Federal regulations).
- Volumes: Cumulative catalog and images up to 50,000 students over 7 years; up to 1.5 million images over 7 years.
- Image size: Accommodate sizes from 4" x 6" to 8.5" x 14"
- Scan process capable of accommodating 2000 scanned images in an 8 hour day, including scanning, cataloging, and dump to storage media. Complete Scan/Catalog time for 1 image needs to be less than 60 seconds (30 or less is preferred).

Taking the Mystery Out of Document Imaging

- Storage: Write an image file to Permanent or temporary storage in less than 15 seconds (10 or less preferred). Retrieve an image file from permanent storage in less than 15 seconds (10 or less preferred).
- Catalog/Index: Store images with retrieval capability by either: SSN or Name (Last, First). Must have logical folders to store documents by financial aid year.
- Image Manipulation: Ability to rotate an image, zoom in on part of an image to 300% or higher, ability to zoom out to 25%, and have 2 images in separate windows open at same time.
- Verify entries to FAMS records: Download list of all student SSNs, names, and birthdates to SQL tables. Keep file up-to-date with new additions to the Financial Aid system. Provide an optional way to build a student's folder not currently on the mainframe file .
- Multiple page document images must be linked together.
- Accommodate SSN changes automatically through a common SSN change file or have the capability to manually move all documents from one SSN to another.


File Zoom Orientation Options Image Help
Pan Window
PELL AWARD
ALREADY
before 1960.
I am a permanent resident of the Federated States of Micronesia, the Marshall Islands or the Republic of Palau.
I certify that I am registered with Selective Service.
Signature
Date
7/2/94

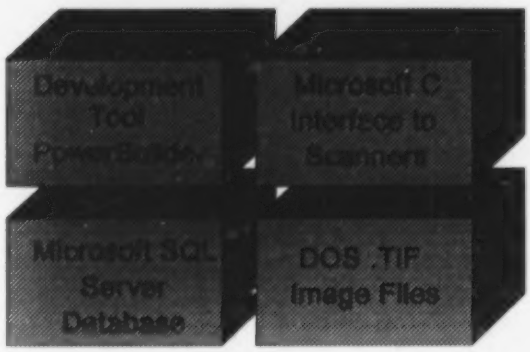
- Ability to correct cataloging errors after-the-fact.
- Must have easy database indicator to denote when it will be OK to purge or archive a student and all their images (after the 7 year period).
- Ability to print an image in less than 30 seconds.
- Document profile must contain the following attributes: SSN, name (Last First), status of student's FAMS file, date of birth, document title, Image page location(s), DRIVE(Optical Disk#), DIRECTORY, FILE NAME(S), # of pages in document, date/time scanned, financial aid year, scan/cataloging person.

An extensive evaluation was done in determining what Windows-based GUI tool to use in developing the application. PowerBuilder was chosen for its strength in 4GL scripting language, building Windows, linking to SQL databases, efficient use of Windows resources, and team development program check-in check-out capabilities. Microsoft SQL Server was selected as the database engine to drive cataloging, indexing, and image tracking. Each team member was given PowerBuilder training.

Interacting with the scanners and image manipulation required doing some programming in C. One individual was dedicated to integrating the C components into the PowerBuilder

application. Dynamic Link Libraries (DLL) were created in C to provide the scanner interface. Image compression, display manipulation, and storage were also developed in C and integrated into the PowerBuilder scripts using external calls to DLLs.

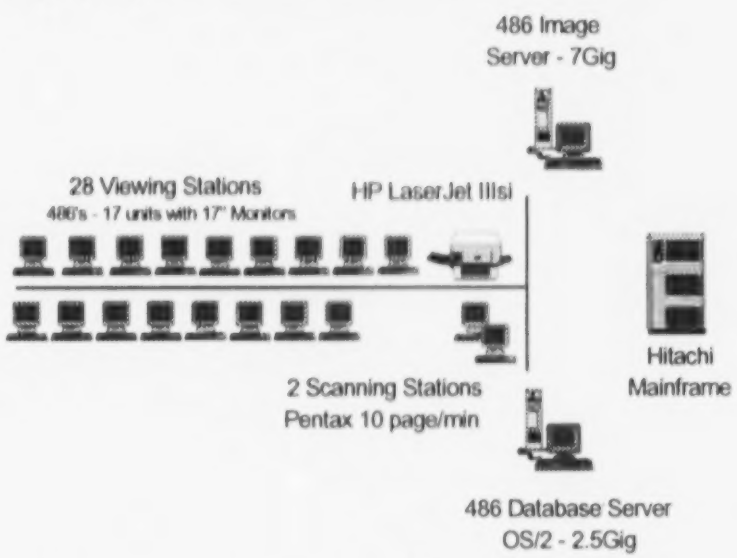
Cataloging and indexing of student documents resides in a SQL Server database on a Banyan OS/2 server. A limited amount of student data is kept on the network SQL Server. Student production data resides on the mainframe in VSAM & ADABAS tables. The system is designed to synchronize new and updated information ported down from the mainframe.



Storing the images posed an interesting challenge. However, the solution was fairly simple. Images are stored in a standard Tag Information Format (TIF) on a DOS drive and are archived in Directories by date. The PowerBuilder application assigns the location and stores that information within the database. The requirements called for fast access and retrieval for documents 2 years old or less. This equated to having approximately 6 to 7 gigabytes of disk on-line. Older documents are to be archived to CD ROM jukebox storage.

Hardware Architecture

The hardware platform consists of 28 IBM 486 viewing stations. Of those 28 viewing stations 17 are equipped with high quality 17" monitors. Each of the desktop units is equipped with 8mb of RAM and 110mb of disk. Two Pentax 10 page per minute scanners are used for all image processing. Printing is done on a HP IIIsi laser printer. A Dell 486 PC is used as the OS/2 database server and contains 2.5 gigabytes of disk. An additional Dell 486 PC is used for storing images and is currently equipped with over 6 gigabytes of disk. The system is to be expanded



BEST COPY AVAILABLE

to 10 or 11 gigabytes of disk within the current year. CD ROM jukebox technology has been chosen for longer term storage. CD ROM offers one of the more cost effective solutions available today. The network topology is Ethernet running Banyan Vines as the Network Operating System connecting to a Hitachi EX27 mainframe system.

Scanning Hardware & Software Costs
Figures do not include existing 28 486 Workstations

Item	Qty	Vendor Cost	Qty	MSCD Costs
Hardware				
Scanning Stations	**1**	3,100	2	6,200
Print Station Board	**1**	1,049		N/A
Jukebox or CD ROM cabinet	**1**	22,932	1	12,969
Image Server	**1**	N/A	1	7,000
Database Server	**1**	24,370	1	7,000
Scanners + Kofax card	**2**	7,410	2	2,886
17" Monitors - Upgrades	**17**	11,900	17	11,900
Software				
Base Application		14,000		N/A
Customization		14,000		N/A
Workstation View Software	**28**	49,000	28	1,680
Scan Station Software	**1**	3,500	2	20
Print Station Software	**1**	3,500		N/A
Database License	**1**	7,000	1	5,195
Installation & Training		10,000		N/A
Jukebox/CD-ROM support software		7,000		2,000
TCP/IP Software for the Mainframe		60,000		N/A
Development Tools	**1**	N/A		7,495
Development Staff				30,000
Total Start up Costs		**238,761**		**94,345**
Savings				**144,416**
Software Cost/additional user		**1,750**		**60**
Yearly Maintenance		**24,024**		**995**

The Financial Aid department implemented the system in the Fall of 1993 and began full scale scanning of student 1995 year financial aid documents in February 1994. Current year document volumes are estimated to be approximately 150,000 single images. The following table indicates scanning progress and documents to date.

Taking the Mystery Out of Document Imaging

Financial Aid - Current Work Status - Through October '94

	Current	Projected
Documents scanned	36081	
Images	70884	150,000/yr
Days images scanned	174	
Average Images per day	407	
Smallest Image size	514 bytes	
Largest Image size	263000 bytes	
Average FTE Required	.5	
Average Image size	42470 bytes	
Total disk usage	3 Gigabytes	7 Gigabytes/yr

The Financial Aid department has already determined they are saving an equivalent of 4.0 FTE normally dedicated to document filing. These FTE, while not eliminated, have been reallocated to other work within the department. Counselors are beginning to use the on-line documents when working with students. This is resulting in improved service to the students and greater individual productivity. Space once filled with boxes of paper has been reclaimed for new offices.

By developing the system internally, the applications development staff gained valuable experience and knowledge in document management, imaging, and workflow applications. This is expertise that would not have been acquired if a vendor solution were procured. The real value of this knowledge will be realized as imaging technologies are extended out to other areas of the College, such as Admissions, Registration, and Transfer Evaluations.

For those considering implementing imaging technologies the following recommendations may be of help.

- **Start small:** Attempting to do too much increases the risk of failure. Keeping the project scope narrow helps focus the project.
- **Don't buy long time storage today:** Disk, Optical, and CD ROM storage costs are dropping fast. It will be cheaper to buy it when you need it.
- **Plan for enough bandwidth in the network:** Providing you select a network solution, be sure the network will handle your projected volume and traffic of documents.
- **Understand how often documents need to be retrieved:** In Financial Aid's case, once a student's loan has been processed, the need to retrieve those documents is rare - keep the documents that have a higher frequency of access on electronic disk.
- **OCR is much better than it has been in the past, however, don't build your system around it yet:** Greater control on the development of outgoing documents

must begin now if OCR is to work for automatic processing of incoming documents.

- **If you go with a vendor package, make sure that it is an Open System:** Some systems use proprietary image compression, databases, and hardware. You may lock yourself into a situation where you can not expand or migrate to another imaging environment. Be leery of proprietary image compression algorithms. It is possible that other packages may not ever be able to read your stored images.
- **If your user does not know how they want to use imaging, do not try to implement workflow on the first try:** One way to tell if they are not ready is by getting conflicting stories on what the current flow of paper is through the office. Introducing workflow into a business environment that is not ready for the changes it brings can be disastrous.
- **Get IS personnel involved early in the process of dealing with vendors, especially if hooks will be needed into existing systems:** Vendors don't know your systems or environment - don't let them tell your users differently.
- **Do a sample shelf-count to get more accurate volume predictions:** Find out if all pages in a document need to be scanned. Determine the volume of two-sided documents.
- **Look closely at your long-term storage needs:** CD-ROM cabinets and cutters may be cheaper and more practical in the long run than Optical Jukeboxes or Magneto-Optical disk.

Conclusion

It has been said that with every significant success there is a significant vision. Undoubtedly the vision for this project was set by the management of Information Technology and supported by the end users. As a result MSCD was successful in developing a document imaging system based upon the existing networked computing environment.

The initial cost savings of $140,000 for the initial system over the proposed vendor solution will be even more dramatic as MSCD deploys imaging to other areas. The incremental cost to extend imaging is approximately $60/workstation compared to $1,750/workstation for the vendor's solution. In addition, recurring maintenance costs of $24,000 were not incurred. The resulting cost savings are greater than originally anticipated. The Financial Aid office has also realized increased productivity and automated efficiencies beyond original expectations. This has all translated to improved service to our students.

The Delphi Consulting Group has estimated an imaging system similar in size and scope to cost approximately $350,000 and take 19 months from RFP to implementation. MSCD's investment of 12 to 13 months in time and $94,000 in costs has shown it is possible to do document imaging cost effectively, without compromising system capabilities or quality.

SIS 2000
The University of Arizona's Team Approach to Client/Server Migration: Strengthening Old Partnerships and Forging New Ones

John Detloff
Elizabeth Taylor
Paul Teitelbaum
Michael Torregrossa
Keith Wilburn

Center for Computing and Information Technology
University of Arizona
Tucson, Arizona 85721

Abstract

At the University of Arizona, we have reached an impasse with our administrative systems. Our mainframe based systems do not adequately support business processes or quality decision making. This paper will outline the approach we, the SIS2000 team, have adopted in launching the University's Student Information System (SIS) into the enabling environment of client/server computing and reengineered processes. Our approach includes marketing our vision, forming and strengthening strategic partnerships, prototyping and implementing new technology, and setting the stage for business process improvement. The campus' positive response to our vision and prototype provides the reengineering effort with the support and political capital it requires to succeed.

VISION

Students

Have you ever...

Applied for admission, completed ALL of your registration tasks, viewed your grades and transcript, communicated with your instructors, checked your financial aid status... using your computer at home, or a computer on campus, at a time that is convenient to you?

Staff

Have you ever...

Had the time to provide quality service to students, instead of having to perform the same repetitive tasks, been able to communicate with colleagues across the campus, the state, and the country, to become more informed and knowledgeable... using a computer system at the office that's as easy to use as the family computer at home?

Faculty

Have you ever...

Managed course offerings, course loads, and course enrollments, assigned final grades, held 'electronic office hours,' communicated with colleagues across campus, the country, and the world... from your computer at the office, at home, or on the road, at convenient times to you?

Administrators

Have you ever...

Viewed accurate, up-to-date teaching load reports, viewed accurate, up-to-date enrollment reports, viewed accurate, up-to-date admissions reports, been able to perform 'what if' simulations using management databases... using your computer at the office, at home, or on the road, at times convenient to you?

You Will!

SIS 2000, helping to launch The University of Arizona into the 21st century!

Borrowed partially from a popular AT&T commercial, the above depicts our vision for transforming the University of Arizona's (UofA) administrative student information system (SIS). We attempted to articulate this vision into a concise, usable statement as follows:

> To achieve a first rate student service and enrollment support system that is easy to use, flexible, and dependable; and is accessible by students, faculty, staff, administrators, and the community, from any place, at any time.

MISSION

We, the SIS2000 team, developed this vision based on a project initiative formulated by Jerry Lucido, assistant vice-president, enrollment services and academic support and

Larry Rapagnani, then the associate vice-president, computing and information technology. SIS applications reside on an IBM 3090 mainframe that is saturated, leading to critical process slowdowns and occasional failures. Upgrades to the system are prohibitively expensive in what continues to be an extremely tight budgetary period. The technology has become dated and no longer adequately supports required enhancements. Jerry Lucido and Larry Rapagnani had the insight to realize that extensive changes were required and initiated the project with the following mission:

> To replace all student system processes that currently reside on the University's 3090 mainframe computer with equivalent, improved and/or reengineered processes, which will reside in a new technological environment utilizing relational database and client/server technologies, and allowing for integration of other related University processes.

With this high-level directive and vision of a new environment in place, we then considered what we needed to do to share and implement our vision. An understanding of the current environment would provide us with information on what needed to be transformed and how that transformation could best take place.

BACKGROUND

The state of affairs at the UofA is not that different than from that at most institutions. We are faced with the challenge of dramatically improving our level of service while facing annual budget cuts of unprecedented amounts. We have business applications based on technologies developed in the 1970's that require major efforts to enhance, if enhancements are possible at all. We have out of date procedures that do not effectively or efficiently support our business that are tied to this technology. And access to information contained in these systems to make quality decisions is, at best, difficult.

Many factors are influencing the demand for improvement. Most of our students are from the "point-n-click" generation. More and more they are expecting the same rapid response from administrative service that they receive from their personal computers. Faculty, staff and administrators are becoming increasingly sophisticated with computer technology and are demanding access to information to effectively perform their duties. Employees that once operated within a strict functional area focus now need to integrate information from many traditional systems.

We identified two concerns that may hinder improvement initiatives. First there is an increasing disparity between those people on campus that have access to, and knowledge of, information resources and those that do not. It is difficult for service providers to improve their business processes when either the technology is not available to them or they are not aware of what it improvements it can provide.

The second concern was that the Center for Computing and Information Technology (CCIT) was often viewed as a provider of specific enhancements to functional area systems and not as a partner in dramatically improving the overall business.

IMPLEMENTING THE VISION

The background analysis emphasized the point that if we wanted to implement our vision, we not only needed to change the technology, but also the organizational culture. We needed to market our vision, educate customers, form strategic partnerships, and successfully demonstrate what level of improvements new technology could support.

The Team

The first step toward implementing the project directive was the formation of the project team. Mike Torregrossa, SIS manager, was chosen as the team leader and determined that to be effective the team needed to contain well-rounded individuals with enthusiasm for implementing improvements. He set the criteria that members have the 'Just Do It!' attitude. Five of us, including Mike, were assigned full-time to the effort. This was a major dedication of resources in an organization already lacking sufficient staffing levels. We believe this demonstrates the seriousness of management toward the reengineering effort.

We chose the name SIS2000 to emphasize that we are looking to the future in solutions to current problems. It is expected that additional team members will join and leave the team as needed, depending on current projects and priorities.

Each team member transferred his or her current duties to other CCIT staff members, whose efforts made the full-time commitment possible. Also, current customers had to agree to lower levels of service. This is certainly a joint effort which requires the cooperation and coordination of many more than just the five of us.

Our first priority was training on becoming an effective team. The training consisted of sessions on Total Quality Management (TQM) concepts such as effective meetings, data gathering, and process analysis as well as learn sessions where individual members shared information on technologies and systems on which they had expertise.

The most important attributes of each team member are his or her communication skills and high level of respect for each other. Each individual brought to the team unique talents ranging from highly technical to process and procedural knowledge. These individual attributes are funneled into group power by using the newly acquired techniques:

- How to hold effective meetings
- How to use group memory systems
- How to use agendas for meetings
- How to use data to make decisions
- How to treat each other as equals and with respect
- How to build consensus about the next steps and future directions
- How to work in very close proximity

During our initial meetings, the team determined that think-tank environment was a criteria for success. We needed an area that would allow us to communicate frequently,

learn from each other, brainstorm on possible solutions and separate us from our traditional duties.

Space is a limited commodity on campus and the only area that met our needs was out of the computer center building. There was some animosity from other employees and at least part of it has been attributed to the separation. We still feel however, that the think-tank environment facilitated much of our success and was a necessity. We are attempting to address the animosity by involving others and communicating more effectively.

We defined the following objectives for the project.

- To improve student and enrollment-related services by providing an information structure supporting the University's Mission and Objectives
- To facilitate an environment where decisions are based on data.
- To create the best higher education administrative information system, where administrators, faculty and staff can concentrate on providing quality service.
- To take advantage of skills and talent, wherever they reside, in an effort to launch the University into the future as the leading land grant institution.

Prototype

After team formation, training and relocation, we began the process of implementing the technical infrastructure to support the reengineering effort. We knew that we had to deliver some quick visible wins to demonstrate our vision and garner the support we needed to implement the levels of change we envisioned.

The University joined in partnership with the Mandarin consortium to take advantage of the infrastructure that was first developed at Cornell University. The University of Arizona aligned with Cornell and 15 other schools to continue the development of Mandarin. This technology helps to maintain the usefulness of legacy data, while easing the transition from mainframe computing to distributed, friendlier technologies. The SIS2000 team enhanced the technology by developing software to operate in the unique environment at the University of Arizona.

We originally estimated that it would take at least six months to implement the infrastructure and provide a proof-of-concept prototype. It was a surprise to everyone, including the team, that this step was accomplished in less than 1 month. Much of the credit for this accelerated time frame was given to the "think tank" environment. The environment provided for instant results that would normally take meetings and scheduling to work out. In addition to the environment, the partnerships with the Mandarin Consortium schools, particularly Cornell University, proved to be invaluable in the implementation phase.

The prototype application's objective was to provide students with access to their own data via an interface that was easy and clear to understand. Initially named UA

Access/Student Access, the prototype system gives students an opportunity to see their addresses, grades, class schedules, financial aid awards, class availability, bills, and other campus information. Through communication and feedback from students the team learned that the students were excited and eagerly awaited this type of technology.

An open house had been planned to introduce the campus community to the new team, their mission, and their new space. When the open house was scheduled, there was not even the dream of being able to show the campus a deliverable product. As mentioned previously, everyone was surprised that on March 17, the day of the open house, four weeks from the date that the team had moved into their new space, a prototype of the technology was available for the campus community to examine and test.

Many people attended the open house: faculty, staff, students and high-level administrators. They saw the potential for new technology and the access it would provide to students. The feedback to the team made it clear that the UA Access prototype should be made available to the students as soon as possible.

The following are examples of UA Access Student menu and the most popular service, class availability: (See figures A and B).

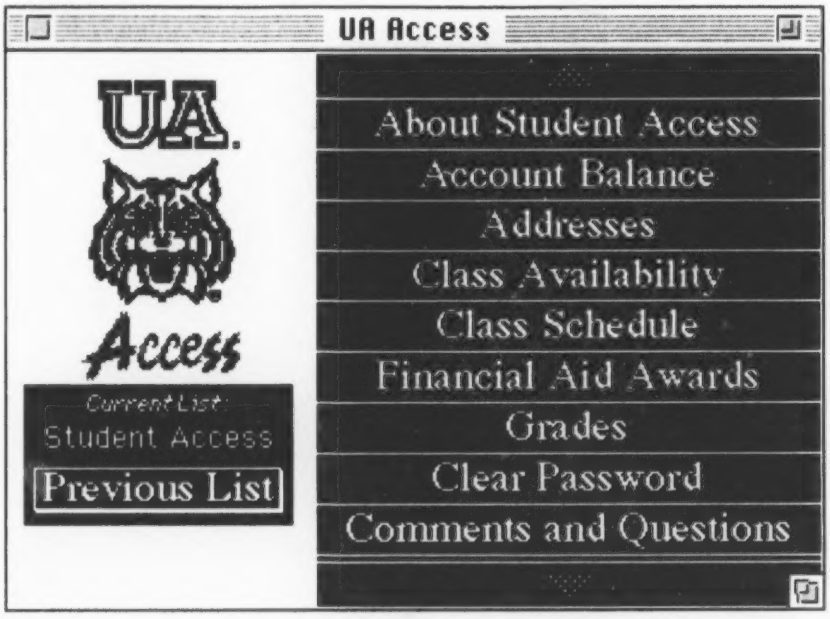


Figure A - Student Access Sub-Menu

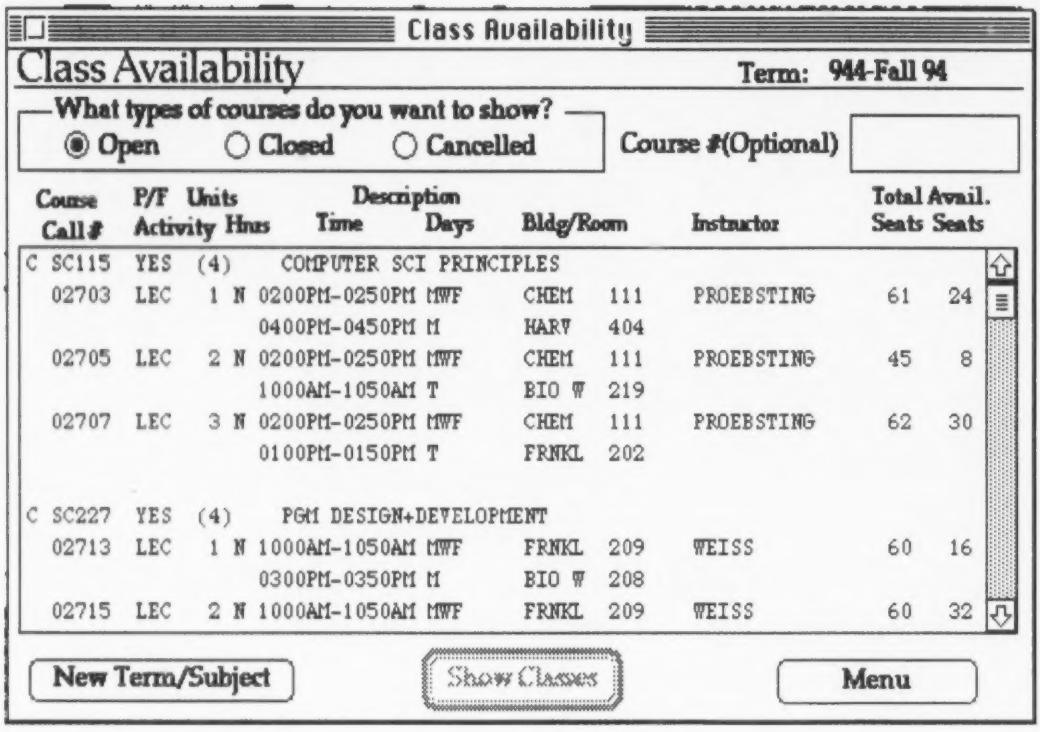

Class Availability
Class Availability
Term: 944-Fall 94
What types of courses do you want to show?
Open
Closed
Cancelled
Course #(Optional)
Course Call # | P/F Activity | Units Hrs | Description Time | Days | Bldg/Room | Instructor | Total Seats | Avail. Seats
C SC115 YES (4) COMPUTER SCI PRINCIPLES
02703 LEC 1 N 0200PM-0250PM MWF CHEM 111 PROEBSTING 61 24
0400PM-0450PM M HARV 404
02705 LEC 2 N 0200PM-0250PM MWF CHEM 111 PROEBSTING 45 8
1000AM-1050AM T BIO W 219
02707 LEC 3 N 0200PM-0250PM MWF CHEM 111 PROEBSTING 62 30
0100PM-0150PM T FRNKL 202
C SC227 YES (4) PGM DESIGN+DEVELOPMENT
02713 LEC 1 N 1000AM-1050AM MWF FRNKL 209 WEISS 60 16
0300PM-0350PM M BIO W 208
02715 LEC 2 N 1000AM-1050AM MWF FRNKL 209 WEISS 60 32
New Term/Subject
Show Classes
Menu

Figure B - Class Availability Service

The following describes services that are currently available to students:

Class Availability - Students can find out which classes are open, closed or canceled. Traditionally, obtaining this information required much time dialing the telephone or waiting in long lines. This is the most popular service for students and parents. The team actually witnessed parents "battling" each other to use UA Access during freshmen orientation!

Grades - Students can look at their own grades for any semester that they had attended classes. This meant that they did not have to wait for the grade cards to come in the mail, or go to the Registrar to find out what grades they received in prior semesters.

Class Schedules - Students can look at their schedules for any semester they have attended the University. This helps them avoid waiting in long lines at registration time, and helps them confirm that they are registered for the correct classes. At the Student Registration Center students may print their schedules, a task traditionally performed by a clerk from the Registrars office.

Financial Aid - Students can see the financial aid they have been awarded. There is no more waiting on award letters, or waiting in line at the Financial Aid office.

Addresses - Students can confirm that they have the correct addresses on file with the University. This ensures the student will get University mailings and reduces return mail costs.

Account Balance- Students can find out how much money they owe or just verify that their balance is zero. They have the option to display billing detail.

Comments and Questions - Students can provide feedback to student service offices. Comments, questions and suggestions can be given anonymously, or they can provide their name, phone number and e-mail address if they request a response.

UAInfo - The new system opened access to the UAInfo system. Students, faculty, and staff can access general campus information on-line. Examples of information include calendars of athletic events, critical deadlines for payments and information about events at the Student Union. UAInfo also provides a direct link to the "information super highway".

Feedback is gathered at the access point as students are using services. A button on the menu labeled "Comments and Questions" allows students, faculty, and staff to type in anything they want. This feedback is electronically mailed directly to members of student service offices and the SIS2000 team where it is categorized and responded to immediately if necessary. This information has been important in mapping the future direction of UA Access and SIS2000. (See figure C for graph of feedback suggestions)

(Figure C - Feedback Suggestions Not Available)

In addition to the feedback data, statistics have been gathered regarding use of UA Access. The following chart shows the usage of each of the student related UA Access services. These usage statistics have helped the team to determine future projects. (See figure D for usage statistics).

(Figure D - Usage Statistics Not Available)

Every student that uses these new services means one less person in line at a student service office. As access to the system grows, employees will be able to utilize their time more effectively by focusing on quality service rather than handling repetitive requests for information.

Marketing

While UA Access was being deployed, our team was busy demonstrating it to administrators, staff, students, and virtually anyone who would take an hour to listen and give feedback. The purpose of the mass marketing process was to gain acceptance and support, and gather feedback for the SIS2000 project.

The biggest, most positive marketing tool was the placement of a kiosk in a visible area within the Student Union. We were stationed at the kiosk during the first week to help market and receive feedback on the product. We received some positive suggestions and implemented some of those quickly. The quick response was another way to show students that this was their service and that we were responsive to their needs.

During this time more than 50 presentations were made to various groups on campus including individual students, student groups, staff, and administrative personnel. One example of how departments became excited by the prototype is after the Financial Aid office saw the demonstration they offered $1000.00 in discretionary scholarship money to involve students in the project. Five hundred dollars was to be awarded for naming the system and five hundred for designing a logo.

When various upper level people saw the demonstrations and the possibilities for improved service, they told us it was time to present our accomplishments to Presidents Cabinet. We shared our vision with and demonstrated our prototype to the cabinet members. The response, from what can be a stoic group, was enthusiastic. Two vice-presidents stood up and volunteered to champion the project. The prototype and marketing had paid off. We now had the political capital and support that we needed to begin the implementation of our vision.

Partnerships

The prototype gave us the support that we needed to proceed, but this project is more than putting another face on legacy systems, it is about dramatically improving service at the University of Arizona. To make the level of improvements that we wanted to make, we needed to define and strengthen many strategic partnerships.

Our partnership with the Mandarin consortium gave us a great lead in providing quick wins, but it also provided us with the opportunity to lay the technical foundation for

reengineering student systems. We have adopted Mandarin technology as a fundamental, strategic factor in our reengineering effort. With the Mandarin partnership, we are able to make use of expertise outside of our organization. Many of the consortium members are trying to solve the same problems that we are.

Partnerships with traditional departmental analysts are strengthening and transforming. Seven analysts from student service offices were added to the initial planning team. The members of this team are: Bill Fee, Curriculum Office, Julie Gardner, Bursar's Office, Lori Goldman, Office of Admissions and New Student Enrollment, Judith Mobasseri, Registrar's Office, Mary Salgado, Registrar's Office, David Soroka, Graduate College, Judy Tran, Office of Student Financial Aid

These team members have been instrumental in developing plans and determining the next steps in the larger re-engineering project. A weekly scheduled meeting with this group has helped to keep the communication lines open and broaden the vision of the team. This group of individuals has the knowledge and experience required to make informed decisions regarding improvement efforts. The relationship with this team is transforming from requesters of specific functional enhancements to full partners in reengineering the business of student service.

Along with cross-departmental partners, "project champions" have volunteered. The role of the project champion includes helping to clear barriers to progress, communicating the project objectives, soliciting funding when needed, and generally becoming involved whenever necessary to keep the momentum going. These individuals have been instrumental in supporting the implementation of the vision. The project champions are: Arlene Becella, registrar, Martha Gilliland, vice provost for academic affairs, Jerry Lucido, assistant vice president for enrollment services, Saundra Taylor, vice president for student affairs

The prototype was implemented in a unique environment with much of the work done outside of the traditional management structure. The success and expansion of the reengineering effort required that a new structure be put in place to guide the project. The new structure, the project management team, was announced by Jerry Lucido and includes representation from campus including students, faculty, staff and administrators. The role of the team is to identify projects, set priorities, and allocate resources for SIS reengineering.

A high-level information plan study was conducted at the University in late summer 1994. The study pointed to the need for the University to rapidly move to a new information environment. The study recommended that administrative systems reengineering be a top priority, with the student system to be addressed first. The study recommended that the University release a Request For Information (RFI), to find out what type of resources a vendor could provide in a potential partnership with the University. Another recommendation of the study was that network connectivity for campus be made a top priority. Implementing the connectivity will help to address the disparity in technical capabilities between campus units and allow departments to partner with the SIS2000 initiative.

The project management team is currently conducting interviews with departments as an extension to the information planning study. These interviews are geared specifically toward student systems. The desired result of the interviews is a definition of customer requirements. The team will also determine if potential vendor partnerships exist for specific projects and they may release an RFI or RFP.

UA Access, originally a prototype, has been put into production and a new team has been formed to expand and improve the service. This team has partnered with another campus group, the UAInfo team. The UAInfo project is responsible for deploying a campus information system that provides on-line access to information such as calendars, phone books, catalogs, etc. The two services are somewhat similar in nature and a partnership would allow them to share limited personnel resources and perhaps make the interfaces of the systems more transparent to campus customers.

In addition to the official supporters, CCIT administrative customers agreed to the lowered service levels that would be provided on the current systems, while team members focused on the new project. This meant understanding that some things would not happen and others would be slower. The removal of five people, three of which worked on the Student Information System, meant many people had to commit to have less service in exchange for future potential and growth.

Technical Architecture

Our team has made extensive use of what was developed at Cornell with the Mandarin consortium. We have enhanced Mandarin to work in the CICS/VSAM environment and are using the technology as a means to begin the SIS migration. We feel that Mandarin is a strategic tool that aids us in laying the technical foundation for reengineering, while providing quick wins to customers. Mandarin technology also allows us to make use of the expertise of existing application developers.

Mandarin is best described in the consortium newsletter as follows: "Project Mandarin is an integrated set of applications, tools, protocols, and procedures that facilitate the development of distributed software systems. The focus is on providing end-user access to data stored in enterprise database systems. The suite of products ranges from stand-alone applications to object-oriented building blocks used for creating custom applications. The suite includes tools used for the creation of graphically oriented client applications, desktop integration of third party applications, and integration of system infrastructure services such as version control, authentication, and authorization. Tools are provided to monitor and maintain the system infrastructure services. There are other tools that provide security, data extraction, and communication with various database systems. Although the products are integrated, they have been developed as modular entities. It is not necessary to implement the entire Project Mandarin suite to benefit from the technology."

We are continuing to expand the Mandarin architecture to provide access to other university data. We are developing both client and server API's so that developers can make use of the technology with minimum knowledge requirements of the Mandarin implementation.

Authentication of people requesting service is a priority for the project. In keeping with the University's standards, we identified DCE Kerberos as a requirement in implementing services. However, Kerberos is still some time away from being fully implemented on campus and we decided not to let it hold up providing some services.

ATTAINING THE VISION

Our team has successfully demonstrated that we can dramatically improve student service. We believe that technology is not the only factor that presents a challenge in improving service, but that the existing organizational culture does as well. With the commitment from individuals within many levels of the organization, the culture is changing as well. Individuals and departments are beginning to recognize each other as strategic partners in improving the business of student service. Increasingly, CCIT is viewed as a critical partner in this improvement effort and not just as a entity for making incremental enhancements to resolve immediate needs.

The prototype defined a vision for campus and garnered the support of students, faculty, administrators, and staff. This support has allowed the SIS2000 team and the University to take progressive steps to the future. As we continue on our path, we intend to remain active in the Mandarin consortium, expand the technology to other administrative systems, and continue to develop strategic partnerships, wherever they exist..

The success of the SIS2000 project continues to require people with initiative, vision and excitement about making a difference.